Sequential design of computer experiments for the estimation of a probability of failure

Julien Bect[⋆] **· David Ginsbourger · Ling Li · Victor Picheny · Emmanuel Vazquez**[⋆]

Received: date / Accepted: date

Abstract This paper deals with the problem of estimating the volume of the excursion set of a function $f : \mathbb{R}^d \to \mathbb{R}$ above a given threshold, under a probability measure on $\mathbb{R}^d$ that is assumed to be known. In the industrial world, this corresponds to the problem of estimating a probability of failure of a system. When only an expensive-to-simulate model of the system is available, the budget for simulations is usually severely limited and therefore classical Monte Carlo methods ought to be avoided. One of the main contributions of this article is to derive *SUR* (*stepwise uncertainty reduction*) strategies from a Bayesian-theoretic formulation of the problem of estimating a probability of failure. These sequential strategies use a Gaussian process model of f and aim at performing evaluations of f as efficiently as possible to infer the value of the probability of failure. We compare these strategies to other strategies also based on a Gaussian process model for estimating a probability of failure.

Keywords Computer experiments · Sequential design · Gaussian processes · Probability of failure · Stepwise uncertainty reduction

1 Introduction

The design of a system or a technological product has to take into account the fact that some design parameters are subject to unknown variations that may affect the reliability of the system. In particular, it is important to estimate the probability of the system to work under abnormal or dangerous operating conditions due to random dispersions of its characteristic parameters. The *probability of failure* of a

⋆ corresponding authors

J. Bect, L. Li, and E. Vazquez
SUPELEC, Gif-sur-Yvette, France.
E-mail: {firstname}.{lastname}@supelec.fr

V. Picheny
Ecole Centrale Paris, Chatenay-Malabry, France.
E-mail: victor.picheny@ecp.fr

D. Ginsbourger
Institute of Mathematical Statistics and Actuarial Science, University of Bern, Switzerland.
E-mail: david.ginsbourger@stat.unibe.ch

system is usually expressed as the probability of the excursion set of a function above a fixed threshold. More precisely, let f be a measurable real function defined over a probability space $(\mathbb{X}, \mathcal{B}(\mathbb{X}), \mathsf{P}_{\mathbb{X}})$, with $\mathbb{X} \subseteq \mathbb{R}^d$, and let $u \in \mathbb{R}$ be a threshold. The problem to be considered in this paper is the estimation of the volume, under $\mathsf{P}_{\mathbb{X}}$, of the excursion set

$$\Gamma := \{x \in \mathbb{X} : f(x) > u\} \tag{1}$$

of the function f above the level u. In the context of robust design, the volume $\alpha := \mathsf{P}_{\mathbb{X}}(\Gamma)$ can be viewed as the probability of failure of a system: the probability $\mathsf{P}_{\mathbb{X}}$ models the uncertainty on the input vector $x \in \mathbb{X}$ of the system—the components of which are sometimes called *design variables* or *factors*—and f is some deterministic performance function derived from the outputs of a deterministic model of the system[1]. The evaluation of the outputs of the model for a given set of input factors may involve complex and time-consuming computer simulations, which turns f into an expensive-to-evaluate function. When f is expensive to evaluate, the estimation of α must be carried out with a *restricted number of evaluations* of f, generally excluding the estimation of the probability of excursion by a Monte Carlo approach. Indeed, consider the empirical estimator

$$\alpha_m := \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}_{\{f(X_i)>u\}} , \tag{2}$$

where the X_is are independent random variables with distribution $\mathsf{P}_{\mathbb{X}}$. According to the strong law of large numbers, the estimator α_m converges to α almost surely when m increases. Moreover, it is an unbiased estimator of α, i.e. $\mathsf{E}(\alpha_m) = \alpha$. Its mean square error is

$$\mathsf{E}\big((\alpha_m - \alpha)^2\big) = \frac{1}{m}\,\alpha\big(1 - \alpha\big) .$$

If the probability of failure α is small, then the standard deviation of α_m is approximately $\sqrt{\alpha/m}$. To achieve a given standard deviation $\delta\alpha$ thus requires approximately $1/(\delta^2\alpha)$ evaluations, which can be prohibitively high if α is small. By way of illustration, if $\alpha = 2 \times 10^{-3}$ and $\delta = 0.1$, we obtain $m = 50000$. If one evaluation of f takes, say, one minute, then the entire estimation procedure will take about 35 days to complete. Of course, a host of refined random sampling methods have been proposed to improve over the basic Monte Carlo convergence rate; for instance, methods based on importance sampling with cross-entropy (Rubinstein and Kroese, 2004), subset sampling (Au and Beck, 2001) or line sampling (Pradlwarter et al., 2007). They will not be considered here for the sake of brevity and because the required number of function evaluations is still very high.

Until recently, all the methods that do not require a large number of evaluations of f were based on the use of parametric approximations for either the function f itself or the boundary $\partial\Gamma$ of Γ. The so-called response surface method falls in the first category (see, e.g., Bucher and Bourgund, 1990; Rajashekhar and Ellingwood, 1993, and references therein). The most popular approaches in the second category are the first- and second-order reliability method (FORM and SORM), which are based on a linear or quadratic approximation of $\partial\Gamma$ around the *most probable failure point* (see, e.g., Bjerager, 1990). In all these methods, the accuracy of the estimator depends on the actual shape of either f or $\partial\Gamma$

[1] Stochastic simulators are also of considerable practical interest, but raise specific modeling and computational issues that will not be considered in this paper.

and its resemblance to the approximant: they do not provide statistically consistent estimators of the probability of failure.

This paper focuses on sequential sampling strategies based on Gaussian processes and kriging, which can been seen as a *non-parametric* approximation method. Several strategies of this kind have been proposed recently in the literature by Ranjan et al. (2008), Bichon et al. (2008), Picheny et al. (2010) and Echard et al. (2010a,b). The idea is that the Gaussian process model, which captures prior knowledge about the unknown function f, makes it possible to assess the uncertainty about the position of Γ given a set of evaluation results. This line of research has its roots in the field of design and analysis of computer experiments (see, e.g., Sacks et al., 1989; Currin et al., 1991; Welch et al., 1992; Oakley and O'Hagan, 2002, 2004; Oakley, 2004; Bayarri et al., 2007). More specifically, kriging-based sequential strategies for the estimation of a probability of failure are closely related to the field of Bayesian global optimization (Mockus et al., 1978; Mockus, 1989; Jones et al., 1998; Villemonteix, 2008; Villemonteix et al., 2009; Ginsbourger, 2009).

The contribution of this paper is twofold. First, we introduce a Bayesian decision-theoretic framework from which the theoretical form of an optimal strategy for the estimation of a probability of failure can be derived. One-step lookahead sub-optimal strategies are then proposed[2], which are suitable for numerical evaluation and implementation on computers. These strategies will be called SUR (stepwise uncertainty reduction) strategies in reference to the work of D. Geman and its collaborators (see, e.g. Fleuret and Geman, 1999). Second, we provide a review in a unified framework of all the kriging-based strategies proposed so far in the literature and compare them numerically with the SUR strategies proposed in this paper.

The outline of the paper is as follows. Section 2 introduces the Bayesian framework and recalls the basics of dynamic programming and Gaussian processes. Section 3 introduces SUR strategies, from the decision-theoretic underpinnings, down to the implementation level. Section 4 provides a review of other kriging-based strategies proposed in the literature. Section 5 provides some illustrations and reports an empirical comparison of these sampling criteria. Finally, Section 6 presents conclusions and offers perspectives for future work.

2 Bayesian decision-theoretic framework

2.1 Bayes risk and sequential strategies

Let f be a continuous function. We shall assume that f corresponds to a computer program whose output is not a closed-form expression of the inputs. Our objective is to obtain a numerical approximation of the probability of failure

$$\alpha(f) = \mathsf{P}_{\mathbb{X}}\{x \in \mathbb{X} : f(x) > u\} = \int_{\mathbb{X}} \mathbb{1}_{f>u} \, \mathrm{d}\mathsf{P}_{\mathbb{X}} \,, \tag{3}$$

where $\mathbb{1}_{f>u}$ stands for the characteristic function of the excursion set Γ, such that for any $x \in \mathbb{X}$, $\mathbb{1}_{f>u}(x)$ equals one if $x \in \Gamma$ and zero otherwise. The approximation of $\alpha(f)$ has to be built from a set of computer experiments, where an experiment simply consists in choosing an $x \in \mathbb{X}$ and computing

[2] Preliminary accounts of this work have been presented in Vazquez and Piera-Martinez (2007) and Vazquez and Bect (2009).

the value of f at x. The result of a pointwise evaluation of f carries information about f and quantities depending on f and, in particular, about $\mathbb{1}_{f>u}$ and $\alpha(f)$. In the context of expensive computer experiments, we shall also suppose that the number of evaluations is limited. Thus, the estimation of $\alpha(f)$ must be carried out using a fixed number, say N, of evaluations of f.

A sequential non-randomized algorithm to estimate $\alpha(f)$ with a budget of N evaluations is a pair $(\underline{X}_N, \widehat{\alpha}_N)$,

$$\underline{X}_N : f \mapsto \underline{X}_N(f) = (X_1(f), X_2(f), \ldots, X_N(f)) \in \mathbb{X}^N\,, \qquad \widehat{\alpha}_N : f \mapsto \widehat{\alpha}_N(f) \in \mathbb{R}_+\,,$$

with the following properties:

a) There exists $x_1 \in \mathbb{X}$ such that $X_1(f) = x_1$, i.e. X_1 does not depend on f.
b) Let $Z_n(f) = f(X_n(f))$, $1 \leq n \leq N$. For all $1 \leq n < N$, $X_{n+1}(f)$ depends measurably[3] on $\mathcal{I}_n(f)$, where $\mathcal{I}_n = ((X_1, Z_1), \ldots, (X_n, Z_n))$.
c) $\widehat{\alpha}_N(f)$ depends measurably on $\mathcal{I}_N(f)$.

The mapping $\underline{X}_N$ will be referred to as a strategy, or policy, or design of experiments, and $\widehat{\alpha}_N$ will be called an estimator. The algorithm $(\underline{X}_N, \widehat{\alpha}_N)$ describes a sequence of decisions, made from an increasing amount of information: $X_1(f) = x_1$ is chosen prior to any evaluation; for each $n = 1, \ldots, N-1$, the algorithm uses information $\mathcal{I}_n(f)$ to choose the next evaluation point $X_{n+1}(f)$; the estimation $\widehat{\alpha}_N(f)$ of $\alpha(f)$ is the terminal decision. In some applications, the class of sequential algorithms must be further restricted: for instance, when K computer simulations can be run in parallel, algorithms that query batches of K evaluations at a time may be preferred (see, e.g. Ginsbourger et al., 2010). In this paper no such restriction is imposed.

The choice of the estimator $\widehat{\alpha}_N$ will be addressed in Section 2.4: for now, we simply assume that an estimator has been chosen, and focus on the problem of finding a good strategy $\underline{X}_N$; that is, one that will produce a good final approximation $\widehat{\alpha}_N(f)$ of $\alpha(f)$. Let $\mathcal{A}_N$ be the class of all strategies $\underline{X}_N$ that query sequentially N evaluations of f. Given a loss function $L : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$, we define the error of approximation of a strategy $\underline{X}_N \in \mathcal{A}_N$ on f as $\epsilon(\underline{X}_N, f) = L(\widehat{\alpha}_N(f), \alpha(f))$. In this paper, we shall consider the quadratic loss function, so that $\epsilon(\underline{X}_N, f) = (\widehat{\alpha}_N(f) - \alpha(f))^2$.

We adopt a Bayesian approach to this decision problem: the unknown function f is considered as a sample path of a real-valued random process ξ defined on some probability space $(\Omega, \mathcal{B}, \mathsf{P}_0)$ with parameter in $x \in \mathbb{X}$, and a good strategy is a strategy that achieves, or gets close to, the *Bayes risk* $r_{\mathrm{B}} := \inf_{\underline{X}_N \in \mathcal{A}_N} \mathsf{E}_0\left(\epsilon(\underline{X}_N, \xi)\right)$, where E_0 denotes the expectation with respect to P_0. From a subjective Bayesian point of view, the stochastic model ξ is a representation of our uncertain initial knowledge about f. From a more pragmatic perspective, the prior distribution can be seen as a tool to define a notion of a good strategy in an average sense. Another interesting route, not followed in this paper, would have been to consider the minimax risk $\inf_{\underline{X}_N \in \mathcal{A}_N} \max_f \mathsf{E}_0\left(\epsilon(\underline{X}_N, \xi)\right)$ over some class of functions.

Notations. From now on, we shall consider the stochastic model ξ instead of the deterministic function f and, for abbreviation, the explicit dependence on ξ will be dropped when no there is no risk of confusion; e.g., $\widehat{\alpha}_N$ will denote the random variable $\widehat{\alpha}_N(\xi)$, X_n will denote the random variable $X_n(\xi)$, etc. We will use the notations $\mathcal{F}_n$, P_n and E_n to denote respectively the σ-algebra generated by $\mathcal{I}_n$, the conditional distribution $\mathsf{P}_0(\,\cdot \mid \mathcal{F}_n)$ and the conditional expectation $\mathsf{E}_0(\,\cdot \mid \mathcal{F}_n)$. Note that

[3] i.e., there is a measurable map $\varphi_n : (\mathbb{X} \times \mathbb{R})^n \to \mathbb{X}$ such that $X_n = \varphi_n \circ \mathcal{I}_n$

the dependence of X_{n+1} on $\mathcal{I}_n$ can be rephrased by saying that X_{n+1} is $\mathcal{F}_n$-measurable. Recall that $\mathsf{E}_n(Z)$ is $\mathcal{F}_n$-measurable, and thus can be seen as a measurable function of $\mathcal{I}_n$, for any random variable Z.

2.2 Optimal and k-step lookahead strategies

It is well-known (see, e.g., Berry and Fristedt, 1985; Mockus, 1989; Bertsekas, 1995) that an optimal strategy for such a finite horizon problem[4], i.e. a strategy $\underline{X}_N^\star \in \mathcal{A}_N$ such that $\mathsf{E}_0\left(\epsilon(\underline{X}_N^\star, \xi)\right) = r_{\mathrm{B}}$, can be formally obtained by *dynamic programming*: let $R_N = \mathsf{E}_N\left(\epsilon(\underline{X}_N, \xi)\right) = \mathsf{E}_N\left((\widehat{\alpha}_N - \alpha)^2\right)$ denote the terminal risk and define by backward induction

$$R_n = \min_{x \in \mathbb{X}} \mathsf{E}_n\big(R_{n+1} \mid X_{n+1} = x\big), \quad n = N-1, \ldots, 0. \tag{4}$$

To get an insight into (4), notice that R_{n+1}, $n = 0, \ldots, N-1$, depends measurably on $\mathcal{I}_{n+1} = (\mathcal{I}_n, X_{n+1}, Z_{n+1})$, so that $\mathsf{E}_n\big(R_{n+1} \mid X_{n+1} = x\big)$ is in fact an expectation with respect to Z_{n+1}, and R_n is an $\mathcal{F}_n$-measurable random variable. Then, we have $R_0 = r_{\mathrm{B}}$, and the strategy $\underline{X}_N^\star$ defined by

$$X_{n+1}^\star = \operatorname*{argmin}_{x \in \mathbb{X}} \mathsf{E}_n\big(R_{n+1} \mid X_{n+1} = x\big), \quad n = 1, \ldots, N-1, \tag{5}$$

is optimal[5]. It is crucial to observe here that, for this dynamic programming problem, both the space of possible actions and the space of possible outcomes at each step are continuous, and the state space $(\mathbb{X} \times \mathbb{R})^n$ at step n is of dimension $n(d+1)$. Any direct attempt at solving (4)–(5) numerically, over an horizon N of more than a few steps, will suffer from the curse of dimensionality.

Using (4), the optimal strategy can be expanded as

$$X_{n+1}^\star = \operatorname*{argmin}_{x \in \mathbb{X}} \mathsf{E}_n\left(\min_{X_{n+2}} \mathsf{E}_{n+1} \ldots \min_{X_N} \mathsf{E}_{N-1} R_N \;\middle|\; X_{n+1} = x\right).$$

A very general approach to construct sub-optimal—but hopefully good—strategies is to truncate this expansion after k terms, replacing the exact risk R_{n+k} by any available surrogate $\widetilde{R}_{n+k}$. Examples of such surrogates will be given in Sections 3 and 4. The resulting strategy,

$$X_{n+1} = \operatorname*{argmin}_{x \in \mathbb{X}} \mathsf{E}_n\left(\min_{X_{n+2}} \mathsf{E}_{n+1} \ldots \min_{X_{n+k}} \mathsf{E}_{n+k-1} \widetilde{R}_{n+k} \;\middle|\; X_{n+1} = x\right). \tag{6}$$

is called a *k-step lookahead strategy* (see, e.g., Bertsekas, 1995, Section 6.3). Note that both the optimal strategy (5) and the k-step lookahead strategy implicitly define a *sampling criterion* $J_n(x)$, $\mathcal{F}_n$-measurable, the minimum of which indicates the next evaluation to be performed. For instance, in the case of the k-step lookahead strategy, the sampling criterion is

$$J_n(x) = \mathsf{E}_n\left(\min_{X_{n+2}} \mathsf{E}_{n+1} \ldots \min_{X_{n+k}} \mathsf{E}_{n+k-1} \widetilde{R}_{n+k} \;\middle|\; X_{n+1} = x\right).$$

[4] in other words, a sequential decision problem where the total number of steps to be performed is known from the start

[5] Proving rigorously that, for a given P_0 and $\widehat{\alpha}_N$, equations (4) and (5) actually define a (measurable!) strategy $\underline{X}_N^\star \in \mathcal{A}_N$ is technical problem that is not of primary interest in this paper. This can be done for instance, in the case of a Gaussian process with continuous covariance function (as considered later), by proving that $x \mapsto \mathsf{E}_n\left(R_{n+1} \mid X_{n+1}(\xi) = x\right)$ is a continuous function on $\mathbb{X}$ and then using a measurable selection theorem.

In the rest of the paper, we restrict our attention to the class of one-step lookahead strategies, which is, as we shall see in Section 3, large enough to provide very efficient algorithms. We leave aside the interesting question of whether more complex k-step lookahead strategies (with $k \geq 2$) could provide a significant improvement over the strategies examined in this paper.

Remark 1 In practice, the analysis of a computer code usually begins with an exploratory phase, during which the output of the code is computed on a *space-filling design* of size $n_0 < N$ (see, e.g., Santner et al., 2003). Such an exploratory phase will be colloquially referred to as the *initial design*. Sequential strategies such as (5) and (6) are meant to be used after this initial design, at steps $n_0 + 1, \ldots, N$. An important (and largely open) question is the choice of the size n_0 of the initial design, for a given global budget N. As a rule of thumb, some authors recommend to start with a sample size proportional to the dimension d of the input space $\mathbb{X}$, for instance $n_0 = 10\,d$; see Loeppky et al. (2009) and the references therein.

2.3 Gaussian process priors

Restricting ξ to be a Gaussian process makes it possible to deal with the conditional distributions P_n and conditional expectations E_n that appear in the strategies above. The idea of modeling an unknown function f by a Gaussian process has originally been introduced circa 1960 in time series analysis (Parzen, 1962), optimization theory (Kushner, 1964) and geostatistics (see, e.g., Chilès and Delfiner, 1999, and the references therein). Today, the Gaussian process model plays a central role in the design and analysis of computer experiments (see, e.g., Sacks et al., 1989; Currin et al., 1991; Welch et al., 1992; Santner et al., 2003). Recall that the distribution of a Gaussian process ξ is uniquely determined by its mean function $m(x) := \mathsf{E}_0(\xi(x))$, $x \in \mathbb{X}$, and its covariance function $k(x, y) := \mathsf{E}_0\left((\xi(x) - m(x))(\xi(y) - m(y))\right)$, $x, y \in \mathbb{X}$. Hereafter, we shall use the notation $\xi \sim \mathrm{GP}\,(m,\, k)$ to say that ξ is a Gaussian process with mean function m and covariance function k.

Let $\xi \sim \mathrm{GP}\,(0,\, k)$ be a zero-mean Gaussian process. The best linear unbiased predictor (BLUP) of $\xi(x)$ from observations $\xi(x_i)$, $i = 1, \ldots, n$, also called the *kriging predictor* of $\xi(x)$, is the orthogonal projection

$$\widehat{\xi}(x; \underline{x}_n) \;:=\; \sum_{i=1}^{n} \lambda_i(x; \underline{x}_n)\, \xi(x_i) \tag{7}$$

of $\xi(x)$ onto $\mathrm{span}\{\xi(x_i), i = 1, \ldots, n\}$. Here, the notation $\underline{x}_n$ stands for the set of points $\underline{x}_n = \{x_1, \ldots, x_n\}$. The weights $\lambda_i(x; \underline{x}_n)$ are the solutions of a system of linear equations

$$k(\underline{x}_n, \underline{x}_n)\lambda(x; \underline{x}_n) = k(x, \underline{x}_n) \tag{8}$$

where $k(\underline{x}_n, \underline{x}_n)$ stands for the $n \times n$ covariance matrix of the observation vector, $\lambda(x; \underline{x}_n) = (\lambda_1(x; \underline{x}_n), \ldots, \lambda_n(x; \underline{x}_n))^{\mathsf{T}}$, and $k(x, \underline{x}_n)$ is a vector with entries $k(x, x_i)$. The function $x \mapsto \widehat{\xi}(x; \underline{x}_n)$ conditioned on $\xi(x_1) = f(x_1), \ldots, \xi(x_n) = f(x_n)$, is deterministic, and provides a cheap *surrogate model* for the true function f (see, e.g., Santner et al., 2003). The covariance function of the error of prediction, also called *kriging covariance* is given by

$$\begin{aligned} k(x, y; \underline{x}_n) \;&:=\; \mathrm{cov}\left(\xi(x) - \widehat{\xi}(x; \underline{x}_n), \xi(y) - \widehat{\xi}(y; \underline{x}_n)\right) \\ &=\; k(x, y) - \sum_i \lambda_i(x; \underline{x}_n)\, k(y, x_i)\,. \end{aligned} \tag{9}$$

The variance of the prediction error, also called the *kriging variance*, is defined as $\sigma^2(x;\underline{x}_n) = k(x,x;\underline{x}_n)$. One fundamental property of a zero-mean Gaussian process is the following (see, e.g., Chilès and Delfiner, 1999, Chapter 3) :

Proposition 1 *If $\xi \sim \mathrm{GP}(0, k)$, then the random process ξ conditioned on the σ-algebra $\mathcal{F}_n$ generated by $\xi(x_1), \ldots, \xi(x_n)$, which we shall denote by $\xi \mid \mathcal{F}_n$, is a Gaussian process with mean $\widehat{\xi}(\,\cdot\,;\underline{x}_n)$ given by (7)-(8) and covariance $k(\,\cdot\,,\cdot\,;\underline{x}_n)$ given by (9). In particular, $\widehat{\xi}(x;\underline{x}_n) = \mathsf{E}_0\big(\xi(x) \mid \mathcal{F}_n\big)$ is the best $\mathcal{F}_n$-measurable predictor of $\xi(x)$, for all $x \in \mathbb{X}$.*

In the domain of computer experiments, the mean of a Gaussian process is generally written as a linear parametric function

$$m(\,\cdot\,) = \beta^{\mathsf{T}} h(\,\cdot\,)\,, \tag{10}$$

where β is a vector of unknown parameters, and $h = (h_1, \ldots, h_l)^{\mathsf{T}}$ is an l-dimensional vector of functions (in practice, polynomials). The simplest case is when the mean function is assumed to be an unknown constant m, in which case we can take $\beta = m$ and $h : x \in \mathbb{X} \mapsto 1$. The covariance function is generally written as a translation-invariant function:

$$k : (x,y) \in \mathbb{X}^2 \mapsto \sigma^2\, \rho_\theta(x-y)\,,$$

where σ^2 is the variance of the (stationary) Gaussian process and ρ_θ is the correlation function, which generally depends on a parameter vector θ. When the mean is written under the form (10), the kriging predictor is again a linear combination of the observations, as in (7), and the weights $\lambda_i(x;\underline{x}_n)$ are again solutions of a system of linear equations (see, e.g., Chilès and Delfiner, 1999), which can be written under a matrix form as

$$\begin{pmatrix} k(\underline{x}_n,\underline{x}_n) & h(\underline{x}_n)^{\mathsf{T}} \\ h(\underline{x}_n) & 0 \end{pmatrix} \begin{pmatrix} \lambda(x;\underline{x}_n) \\ \mu(x) \end{pmatrix} = \begin{pmatrix} k(x,\underline{x}_n) \\ h(x) \end{pmatrix}, \tag{11}$$

where $h(\underline{x}_n)$ is an $l \times n$ matrix with entries $h_i(x_j)$, $i = 1, \ldots, l$, $j = 1, \ldots, n$, μ is a vector of Lagrange coefficients ($k(\underline{x}_n,\underline{x}_n)$, $\lambda(x;\underline{x}_n)$, $k(x,\underline{x}_n)$ as above). The kriging covariance function is given in this case by

$$\begin{aligned} k(x,y;\underline{x}_n) &:= \operatorname{cov}\Big(\xi(x) - \widehat{\xi}(x;\underline{x}_n), \xi(y) - \widehat{\xi}(y;\underline{x}_n)\Big) \\ &= k(x,y) - \lambda(x;\underline{x}_n)^{\mathsf{T}}\, k(y,\underline{x}_n) - \mu(x)^{\mathsf{T}} h(y)\,. \end{aligned} \tag{12}$$

The following result holds (Kimeldorf and Wahba, 1970; O'Hagan, 1978):

Proposition 2 *Let k be a covariance function.*

$$\textit{If}\ \begin{cases} \xi \mid m \ \sim\ \mathrm{GP}(m,\, k) \\ m : x \mapsto \beta^{\mathsf{T}} h(x),\ \beta \ \sim\ \mathcal{U}_{\mathbb{R}^l} \end{cases} \quad \textit{then}\ \ \xi \mid \mathcal{F}_n \ \sim\ \mathrm{GP}\Big(\widehat{\xi}(\,\cdot\,;\underline{x}_n),\, k(\,\cdot\,,\,\cdot\,;\underline{x}_n)\Big)\,,$$

where $\mathcal{U}_{\mathbb{R}^l}$ stands for the (improper) uniform distribution over $\mathbb{R}^l$, and where $\widehat{\xi}(\,\cdot\,;\underline{x}_n)$ and $k(\,\cdot\,,\,\cdot\,;\underline{x}_n)$ are given by (7), (11) and (12).

Proposition 2 justifies the use of kriging in a Bayesian framework provided that the covariance function of ξ is known. However, the covariance function is rarely assumed to be known in applications. Instead, the covariance function is generally taken in some parametric class (in this paper, we use the so-called

Matérn covariance function, see Appendix A). A *fully Bayesian* approach also requires to choose a prior distribution for the unknown parameters of the covariance (see, e.g., Handcock and Stein, 1993; Kennedy and O'Hagan, 2001; Paulo, 2005). Sampling techniques (Monte Carlo Markov Chains, Sequential Monte Carlo...) are then generally used to approximate the posterior distribution of the unknown covariance parameters. Very often, the popular *empirical Bayes* approach is used instead, which consists in plugging-in the maximum likelihood (ML) estimate to approximate the posterior distribution of ξ. This approach has been used in previous papers about contour estimation or probability of failure estimation (Picheny et al., 2010; Ranjan et al., 2008; Bichon et al., 2008). In Section 5.2 we will adopt a plug-in approach as well.

Simplified notations. In the rest of the paper, we shall use the following simplified notations when there is no risk of confusion: $\widehat{\xi}_n(x) := \widehat{\xi}(x; \underline{X}_n)$, $\sigma_n^2(x) := \sigma^2(x; \underline{X}_n)$.

2.4 Estimators of the probability of failure

Given a random process ξ and a strategy $\underline{X}_N$, the optimal estimator that minimizes $\mathsf{E}_0\left((\alpha - \widehat{\alpha}_n)^2\right)$ among all $\mathcal{F}_n$-measurable estimators $\widehat{\alpha}_n$, $1 \leq n \leq N$, is

$$\widehat{\alpha}_n \;=\; \mathsf{E}_n\left(\alpha\right) \;=\; \mathsf{E}_n\left(\int_{\mathbb{X}} \mathbb{1}_{\xi>u}\, \mathrm{d}\mathsf{P}_{\mathbb{X}}\right) \;=\; \int_{\mathbb{X}} p_n\, \mathrm{d}\mathsf{P}_{\mathbb{X}}\,, \tag{13}$$

where

$$p_n : x \in \mathbb{X} \mapsto \mathsf{P}_n\left\{\xi(x) > u\right\}\,. \tag{14}$$

When ξ is a Gaussian process, the probability $p_n(x)$ of exceeding u at $x \in \mathbb{X}$ given $\mathcal{I}_n$ has a simple closed-form expression:

$$p_n(x) \;=\; 1 \,-\, \Phi\left(\frac{u - \widehat{\xi}_n(x)}{\sigma_n(x)}\right) \;=\; \Phi\left(\frac{\widehat{\xi}_n(x) - u}{\sigma_n(x)}\right)\,, \tag{15}$$

where Φ is the cumulative distribution function of the normal distribution. Thus, in the Gaussian case, the estimator (13) is amenable to a numerical approximation, by integrating the excess probability p_n over $\mathbb{X}$ (for instance using Monte Carlo sampling, see Section 3.3).

Another natural way to obtain an estimator of α given $\mathcal{I}_n$ is to approximate the excess indicator $\mathbb{1}_{\xi>u}$ by a hard classifier $\eta_n : \mathbb{X} \to \{0, 1\}$, where "hard" refers to the fact that η_n takes its values in $\{0, 1\}$. If η_n is close in some sense to $\mathbb{1}_{\xi>u}$, the estimator

$$\widehat{\alpha}_n = \int_{\mathbb{X}} \eta_n \mathrm{d}\mathsf{P}_{\mathbb{X}} \tag{16}$$

should be close to α. More precisely,

$$\mathsf{E}_n\left((\widehat{\alpha}_n - \alpha)^2\right) = \mathsf{E}_n\left[\left(\int (\eta_n - \mathbb{1}_{\xi>u}) \mathrm{d}\mathsf{P}_{\mathbb{X}}\right)^2\right] \leq \int \mathsf{E}_n\left((\eta_n - \mathbb{1}_{\xi>u})^2\right) \mathrm{d}\mathsf{P}_{\mathbb{X}}\,. \tag{17}$$

Let $\tau_n(x) = \mathsf{P}_n\{\eta_n(x) \neq \mathbb{1}_{\xi(x)>u}\} = \mathsf{E}_n\left((\eta_n(x) - \mathbb{1}_{\xi(x)>u})^2\right)$ be the probability of misclassification; that is, the probability to predict a point above (resp. under) the threshold when the true value is under (resp. above) the threshold. Thus, (17) shows that it is desirable to use a classifier η_n such that τ_n is

small for all $x \in \mathbb{X}$. For instance, the method called SMART (Deheeger and Lemaire, 2007) uses a support vector machine to build η_n. Note that

$$\tau_{n(x)} \;=\; p_n(x) \,+\, (1 - 2p_n(x))\, \eta_n(x)\,.$$

Therefore, the right-hand side of (17) is minimized if we set

$$\eta_n(x) = \mathbb{1}_{p_n(x)>1/2} = \mathbb{1}_{\bar{\xi}_n(x)>u}\,, \tag{18}$$

where $\bar{\xi}_n(x)$ denotes the posterior median of $\xi(x)$. Then, we have

$$\tau_n(x) = \min(p_n(x), 1 - p_n(x)).$$

In the case of a Gaussian process, the posterior median and the posterior mean are equal. Then, the classifier that minimizes $\tau_n(x)$ for each $x \in \mathbb{X}$ is $\eta_n = \mathbb{1}_{\widehat{\xi}_n>u}$, in which case

$$\tau_n(x) \;=\; \mathsf{P}_n\left((\xi(x) - u)(\widehat{\xi}_n(x) - u) < 0\right) \;=\; 1 \,-\, \Phi\left(\frac{\left|\widehat{\xi}_n(x) - u\right|}{\sigma_n(x)}\right)\,. \tag{19}$$

Notice that for $\eta_n = \mathbb{1}_{\widehat{\xi}_n>u}$, we have $\widehat{\alpha}_n = \alpha(\widehat{\xi}_n)$. Therefore, this approach to obtain an estimator of α can be seen as a type of plug-in estimation.

Standing assumption. It will assumed in the rest of the paper that ξ is a Gaussian process, or more generally that $\xi \mid \mathcal{F}_n \sim \mathrm{GP}\big(\widehat{\xi}_n,\, k(\cdot\,,\cdot\,;\underline{x}_n)\big)$ for all $n \geq 1$ as in Proposition 2.

3 Stepwise uncertainty reduction

3.1 Principle

A very natural and straightforward way of building a one-step lookahead strategy is to select *greedily* each evaluation as if it were the last one. This kind of strategy, sometimes called *myopic*, has been successfully applied in the field of Bayesian global optimization (Mockus et al., 1978; Mockus, 1989), yielding the famous *expected improvement* criterion later popularized in the Efficient Global Optimization (EGO) algorithm of Jones et al. (1998).

When the Bayesian risk provides a measure of the estimation error or uncertainty (as in the present case), we call such a strategy a *stepwise uncertainty reduction* (SUR) strategy. In the field of global optimization, the Informational Approach to Global Optimization (IAGO) of Villemonteix et al. (2009) is an example of a SUR strategy, where the Shannon entropy of the minimizer is used instead of the quadratic cost. When considered in terms of utility rather than cost, such strategies have also been called *knowledge gradient policies* by Frazier et al. (2008).

Given a sequence of estimators $(\widehat{\alpha}_n)_{n\geq 1}$, a direct application of the above principle using the quadratic loss function yields the sampling criterion

$$J_n(x) = \mathsf{E}_n\left((\alpha - \widehat{\alpha}_{n+1})^2 \mid X_{n+1} = x\right)\,. \tag{20}$$

Having found no closed-form expression for this criterion, and no efficient numerical procedure for its approximation, we will proceed by upper-bounding and discretizing (20) in order to get an expression that will lend itself to a numerically tractable approximation. By doing so, several SUR strategies will be derived, depending on the choice of estimator (the posterior mean (13) or the plug-in estimator (16) with (18)) and bounding technique.

3.2 Upper bounds of the SUR sampling criterion

Recall that $\tau_n(x) = \min(p_n(x), 1-p_n(x))$ is the probability of misclassification at x using the optimal classifier $\mathbb{1}_{\widehat{\xi}_n(x)>u}$. Let us further denote by $\nu_n(x) := p_n(x)\,(1-p_n(x))$ the variance of the excess indicator $\mathbb{1}_{\xi(x)\geq u}$.

Proposition 3 *Assume that either* $\widehat{\alpha}_n = \mathsf{E}_n(\alpha)$ *or* $\widehat{\alpha}_n = \int \mathbb{1}_{\widehat{\xi}_n \geq u} \mathrm{dP}_{\mathbb{X}}$. *Define* $G_n := \int_{\mathbb{X}} \sqrt{\gamma_n(y)} \mathrm{dP}_{\mathbb{X}}$ *for all* $n \in \{0, \ldots, N-1\}$, *with*

$$\gamma_n := \begin{cases} \nu_n = p_n(1-p_n) = \tau_n(1-\tau_n)\,, & \text{if } \widehat{\alpha}_n = \mathsf{E}_n(\alpha)\,, \\ \tau_n = \min(p_n, 1-p_n)\,, & \text{if } \widehat{\alpha}_n = \int \mathbb{1}_{\widehat{\xi}_n \geq u} \mathrm{dP}_{\mathbb{X}}\,. \end{cases}$$

Then, for all $x \in \mathbb{X}$ *and all* $n \in \{0, \ldots, N-1\}$,

$$J_n(x) \;\leq\; \widetilde{J}_n(x) \;:=\; \mathsf{E}_n\left(G_{n+1}^2 \mid X_{n+1} = x\right).$$

Note that $\gamma_n(x)$ is a function of $p_n(x)$ that vanishes at 0 and 1, and reaches its maximum at 1/2; that is, when the uncertainty on $\mathbb{1}_{\widehat{\xi}_n(x)>u}$ is maximal (see Figure 1).

Proof First, observe that, for all $n \geq 0$, $\alpha - \widehat{\alpha}_n = \int U_n \,\mathrm{dP}_{\mathbb{X}}$, with

$$U_n : x \in \mathbb{X} \mapsto U_n(x) \;=\; \begin{cases} \mathbb{1}_{\xi(x)>u} - p_n(x) & \text{if } \widehat{\alpha}_n = \mathsf{E}_n(\alpha)\,, \\ \mathbb{1}_{\xi(x)>u} - \mathbb{1}_{\widehat{\xi}_n(x)>u} & \text{if } \widehat{\alpha}_n = \int \mathbb{1}_{\widehat{\xi}_n \geq u} \mathrm{dP}_{\mathbb{X}}\,. \end{cases} \tag{21}$$

Moreover, note that $\gamma_n = \|U_n\|_n^2$ in both cases, where $\|\cdot\|_n : L^2(\Omega, \mathcal{B}, \mathsf{P}) \to L^2(\Omega, \mathcal{F}_n, \mathsf{P})$, $W \mapsto \mathsf{E}_n\big(W^2\big)^{1/2}$. Then, using the generalized Minkowski inequality (see, e.g., Vestrup, 2003, section 10.7) we get that

$$\left\| \int U_n \,\mathrm{dP}_{\mathbb{X}} \right\|_n \;\leq\; \int \|U_n\|_n \,\mathrm{dP}_{\mathbb{X}} \;=\; \int \sqrt{\gamma_n} \,\mathrm{dP}_{\mathbb{X}} \;=\; G_n. \tag{22}$$

Finally, it follows from the tower property of conditional expectations and (22) that, for all $n \geq 0$,

$$\begin{aligned} J_n(x) \;&= \mathsf{E}_n\left(\|\alpha - \widehat{\alpha}_{n+1}\|_{n+1}^2 \mid X_{n+1} = x\right) \\ &= \mathsf{E}_n\left(\left\|\int U_{n+1} \,\mathrm{dP}_{\mathbb{X}}\right\|_{n+1}^2 \,\middle|\, X_{n+1} = x\right) \\ &\leq \mathsf{E}_n\left(G_{n+1}^2 \mid X_{n+1} = x\right). \end{aligned}$$

□

Note that two other upper-bounding sampling criteria readily follow from those of Proposition 3, by using the Cauchy-Schwarz inequality in $L^2(\mathbb{X}, \mathcal{B}(\mathbb{X}), \mathsf{P}_{\mathbb{X}})$:

$$\widetilde{J}_n(x) \;\leq\; \mathsf{E}_n\left(\int \gamma_{n+1} \,\mathrm{dP}_{\mathbb{X}} \,\middle|\, X_{n+1} = x\right). \tag{23}$$

As a result, we can write four SUR criteria, whose expressions are summarized in Table 1. Criterion $J_{1,n}^{\mathrm{SUR}}$ has been proposed in the PhD thesis of Piera-Martinez (2008) and in conference papers (Vazquez and Piera-Martinez, 2007; Vazquez and Bect, 2009); the other ones, to the best of our knowledge, are new. Each criterion is expressed as the conditional expectation of some (possibly squared) $\mathcal{F}_{n+1}$-measurable integral criterion, with an integrand that can be expressed as a function of the probability

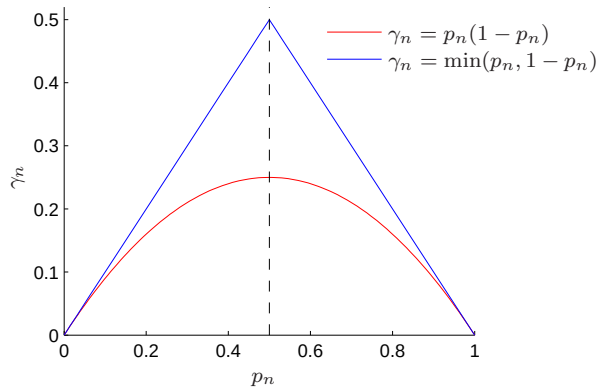

Fig. 1 γ_n as a function of p_n (see Proposition 3). In both cases, γ_n is maximum at $p_n = 1/2$.

of misclassification τ_{n+1}. It is interesting to note that the integral in J_4^{SUR} is the integrated mean square error (IMSE)[6] for the process $\mathbb{1}_{\xi>u}$.

Remark 2 The conclusions of Proposition 3 still hold in the general case when ξ is not assumed to be a Gaussian process, provided that the posterior median $\bar{\xi}_n$ is substituted to posterior the mean $\widehat{\xi}_n$.

Table 1 Expressions of four SUR-type criteria.

SUR-type sampling criterion	How it is obtained
$J_{1,n}^{\mathrm{SUR}}(x) = \mathsf{E}_n\Big(\big(\int \sqrt{\tau_{n+1}}\,\mathrm{dP}_{\mathbb{X}}\big)^2 \;\Big|\; X_{n+1} = x \Big)$	Prop. 3 with $\widehat{\alpha}_n = \int \mathbb{1}_{\widehat{\xi}_n>u}\,\mathrm{dP}_{\mathbb{X}}$
$J_{2,n}^{\mathrm{SUR}}(x) = \mathsf{E}_n\Big(\big(\int \sqrt{\nu_{n+1}}\,\mathrm{dP}_{\mathbb{X}}\big)^2 \;\Big|\; X_{n+1} = x \Big)$	Prop. 3 with $\widehat{\alpha}_n = \mathsf{E}_n(\alpha)$
$J_{3,n}^{\mathrm{SUR}}(x) = \mathsf{E}_n\Big(\int \tau_{n+1}\,\mathrm{dP}_{\mathbb{X}} \;\Big|\; X_{n+1} = x \Big)$	Eq. (23) with $\widehat{\alpha}_n = \int \mathbb{1}_{\widehat{\xi}_n>u}\,\mathrm{dP}_{\mathbb{X}}$
$J_{4,n}^{\mathrm{SUR}}(x) = \mathsf{E}_n\Big(\int \nu_{n+1}\,\mathrm{dP}_{\mathbb{X}} \;\Big|\; X_{n+1} = x \Big)$	Eq. (23) with $\widehat{\alpha}_n = \mathsf{E}_n(\alpha)$

3.3 Discretizations

In this section, we proceed with the necessary integral discretizations of the SUR criteria to make them suitable for numerical evaluation and implementation on computers. Assume that n steps of the algorithm have already been performed and consider, for instance, the criterion

$$J_{3,n}^{\mathrm{SUR}}(x) = \mathsf{E}_n\Big(\int \tau_{n+1}(y)\,\mathsf{P}_{\mathbb{X}}(\mathrm{d}y) \;\Big|\; X_{n+1} = x \Big). \tag{24}$$

Remember that, for each $y \in \mathbb{X}$, the probability of misclassification $\tau_{n+1}(y)$ is $\mathcal{F}_{n+1}$-measurable and, therefore, is a function of $\mathcal{I}_{n+1} = (\mathcal{I}_n, X_{n+1}, Z_{n+1})$. Since $\mathcal{I}_n$ is known at this point, we introduce the

[6] The IMSE criterion is usually applied to the response surface ξ itself (see, e.g., Box and Draper, 1987; Sacks et al., 1989). The originality here is to consider the IMSE of the process $\mathbb{1}_{\xi>u}$ instead. Another way of adapting the IMSE criterion for the estimation of a probability of failure, proposed by Picheny et al. (2010), is recalled in Section 4.2.

notation $v_{n+1}(y; X_{n+1}, Z_{n+1}) = \tau_{n+1}(y)$ to emphasize the fact that, when a new evaluation point must be chosen at step $(n+1)$, $\tau_{n+1}(y)$ depends on the choice of X_{n+1} and the random outcome Z_{n+1}. Let us further denote by $\mathsf{Q}_{n,x}$ the probability distribution of $\xi(x)$ under P_n. Then, (24) can be rewritten as

$$J_{3,n}^{\mathrm{SUR}}(x) = \iint_{\mathbb{R}\times\mathbb{X}} v_{n+1}(y; x, z)\, \mathsf{Q}_{n,x}(\mathrm{d}z)\, \mathsf{P}_{\mathbb{X}}(\mathrm{d}y)\,,$$

and the corresponding strategy is:

$$X_{n+1} \;=\; \operatorname*{argmin}_{x\in\mathbb{X}} \iint_{\mathbb{R}\times\mathbb{X}} v_{n+1}(y; x, z)\, \mathsf{Q}_{n,x}(\mathrm{d}z)\, \mathsf{P}_{\mathbb{X}}(\mathrm{d}y)\,. \tag{25}$$

Given $\mathcal{I}_n$ and a triple (x, y, z), $v_{n+1}(y; x, z)$ can be computed efficiently using the equations provided in Sections 2.3 and 2.4.

At this point, we need to address: 1) the computation of the integral on $\mathbb{X}$ with respect to $\mathsf{P}_{\mathbb{X}}$; 2) the computation of the integral on $\mathbb{R}$ with respect to $\mathsf{Q}_{n,x}$; 3) the minimization of the resulting criterion with respect to $x \in \mathbb{X}$.

To solve the first problem, we draw an i.i.d. sequence $Y_1, \ldots, Y_m \sim \mathsf{P}_{\mathbb{X}}$ and use the Monte Carlo approximation:

$$\int_{\mathbb{X}} v_{n+1}(y; x, z)\, \mathsf{P}_{\mathbb{X}}(\mathrm{d}y) \;\approx\; \frac{1}{m}\sum_{j=1}^{m} v_{n+1}(Y_j; x, z).$$

An increasing sample size $n \mapsto m_n$ should be used to build a convergent algorithm for the estimation of α (possibly with a different sequence $Y_{n,1}, \ldots, Y_{n,m_n}$ at each step). In this paper we adopt a different approach instead, which is to take a fixed sample size $m > 0$ and keep the same sample $Y_1, \ldots, Y_m$ throughout the iterations. Equivalently, it means that we choose to work from the start on a discretized version of the problem: we replace $\mathsf{P}_{\mathbb{X}}$ by the empirical distribution $\widehat{\mathsf{P}}_{\mathbb{X},n} = \frac{1}{m}\sum_{j=1}^{m} \delta_{Y_j}$, and our goal is now to *estimate the Monte Carlo estimator* $\alpha_m = \int \mathbb{1}_{\xi>u} \mathrm{d}\widehat{\mathsf{P}}_{\mathbb{X},n} = \frac{1}{m}\sum_{j=1}^{m} \mathbb{1}_{\xi(Y_j)>u}$, using either the posterior mean $\mathsf{E}_n(\alpha_m) = \frac{1}{m}\sum_j p_n(Y_j)$ or the plug-in estimate $\frac{1}{m}\sum_j \mathbb{1}_{\widehat{\xi}(Y_j;\underline{X}_n)>u}$. This kind of approach has be coined *meta-estimation* by Arnaud et al. (2010): the objective is to estimate the value of a precise Monte Carlo estimator of $\alpha(f)$ (m being large), using prior information on f to alleviate the computational burden of running m times the computer code f. This point of view also underlies the work in structural reliability of Hurtado (2004, 2007), Deheeger and Lemaire (2007), Deheeger (2008), and more recently Echard et al. (2010a,b).

The new point of view also suggests a natural solution for the third problem, which is to replace the continuous search for a minimizer $x \in \mathbb{X}$ by a discrete search over the set $\mathbb{X}_m := \{Y_1, \ldots, Y_m\}$. This is obviously sub-optimal, even in the meta-estimation framework introduced above, since picking $x \in \mathbb{X}\setminus\mathbb{X}_m$ can sometimes bring more information about $\xi(Y_1), \ldots, \xi(Y_m)$ than the best possible choice in $\mathbb{X}_m$. Global optimization algorithms may of course be used to tackle directly the continuous search problem: for instance, Ranjan et al. (2008) use a combination of a genetic algorithm and local search technique, Bichon et al. (2008) use the DIRECT algorithm and Picheny et al. (2010) use a covariance-matrix-adaptation evolution strategy. In this paper we will stick to the discrete search approach, since it is much simpler to implement (we shall present in Section 3.4 a method to handle the case of large m) and provides satisfactory results (see Section 5).

Finally, remark that the second problem boils down to the computation of a one-dimensional integral with respect to Lebesgue's measure. Indeed, since ξ is a Gaussian process, $\mathsf{Q}_{n,x}$ is a Gaussian probability

distribution with mean $\widehat{\xi}_n(x)$ and variance $\sigma_n^2(x)$ as explained in Section 2.3. The integral can be computed using a standard Gauss-Hermite quadrature with Q points (see, e.g., Press et al., 1992, Chapter 4) :

$$\int v_{n+1}(y;x,z)\,\mathsf{Q}_{n,x}(\mathrm{d}z) \;\approx\; \frac{1}{\sqrt{\pi}}\sum_{q=1}^{Q} w_q\, v_{n+1}(y;x,\widehat{\xi}_n(x)+\sigma_n(x)u_q\sqrt{2})\,,$$

where $u_1,\ldots,u_Q$ denote the quadrature points and $w_1,\ldots,w_Q$ the corresponding weights. Note that this is equivalent to replacing under P_n the random variable $\xi(x)$ by a quantized random variable with probability distribution $\sum_{q=1}^{Q} w'_q\delta_{z_{n+1,q}(x)}$, where $w'_q = w_q/\sqrt{\pi}$ and $z_{n+1,q}(x) = \widehat{\xi}_n(x)+\sigma_n(x)u_q\sqrt{2}$.

Taking all three discretizations into account, the proposed strategy is:

$$X_{n+1} \;=\; \operatorname*{argmin}_{1\le k\le m} \sum_{j=1}^{m}\sum_{q=1}^{Q} w'_q\, v_{n+1}\left(Y_j;\, Y_k, z_{n+1,q}(Y_k)\right). \tag{26}$$

3.4 Implementation

This section gives implementation guidelines for the SUR strategies described in Section 3. As said in Section 3.3, the strategy (26) can, in principle, be translated directly into a computer program. In practice however, we feel that there is still room for different implementations. In particular, it is important to keep the computational complexity of the strategies at a reasonable level. We shall explain in this section some simplifications we have made to achieve this goal.

A straight implementation of (26) for the choice of an additional evaluation point is described in Table 2. This procedure is meant to be called iteratively in a sequential algorithm, such as that described for instance in Table 3. Note that the only parameter to be specified in the SUR strategy (26) is Q, which tunes the precision of the approximation of the integral on $\mathbb{R}$ with respect to $\mathsf{Q}_{n,x}$. In our numerical experiments, it was observed that taking $Q = 12$ achieves a good compromise between precision and numerical complexity.

To assess the complexity of a SUR sampling strategy, recall that kriging takes $\mathrm{O}(mn^2)$ operations to predict the value of f at m locations from n evaluation results of f (we suppose that $m > n$ and no approximation is carried out). In the procedure to select an evaluation, a first kriging prediction is performed at Step 1 and then, m different predictions have to performed at step 2.1. This cost becomes rapidly burdensome for large values of n and m, and we must further simplify (26) to be able to work on applications where m must be large. A natural idea to alleviate the computational cost of the strategy is to avoid dealing with candidate points that have a very low probability of misclassification, since they are probably far from the frontier of the domain of failure. It is also likely that those points with a low probability of misclassification will have a very small contribution in the variance of the error of estimation $\widehat{\alpha}_n - \alpha_m$.

Therefore, the idea is to rewrite the sampling strategy described by (26), in such a way that the first summation (over m) and the search set for the minimizer is restricted to a subset of points Y_j corresponding to the m_0 largest values of $\tau_n(Y_j)$. The corresponding algorithm is not described here for the sake of brevity but can easily be adapted from that of Table 2. Sections 5.2 and 5.3 will show that this *pruning* scheme has almost no consequence on the performances of the SUR strategies, even when one considers small values for m_0 (for instance $m_0 = 200$).

Table 2 Procedure to select a new evaluation point $X_{n+1} \in \mathbb{X}$ using a SUR strategy

Require computer representations of

a) a set $\mathcal{I}_n = \{(X_1, f(X_1)), \ldots, (X_n, f(X_n))\}$ of evaluation results;
b) a Gaussian process prior ξ with a (possibly unknown linear parametric) mean function and a covariance function k_θ, with parameter θ;
c) a (pseudo-)random sample $\mathbb{X}_m = \{Y_1, \ldots, Y_m\}$ of size m drawn from the distribution $\mathsf{P}_{\mathbb{X}}$;
d) quadrature points $u_1, \ldots, u_Q$ and corresponding weights $w'_1, \ldots, w'_Q$;
e) a threshold u.

1. compute the kriging approximation $\widehat{f}_n$ and kriging variance σ_n^2 on $\mathbb{X}_m$ from $\mathcal{I}_n$
2. for each candidate point Y_j, $j \in \{1, \ldots, m\}$,
   2.1 for each point Y_k, $k \in \{1, \ldots, m\}$, compute the kriging weights $\lambda_i(Y_k; \{\underline{X}_n, Y_j\})$, $i \in \{1, \ldots, (n+1)\}$, and the kriging variances $\sigma^2(Y_k; \{\underline{X}_n, Y_j\})$
   2.2 compute $z_{n+1,q}(Y_j) = \widehat{f}_n(Y_j) + \sigma_n(Y_j) u_q \sqrt{2}$, for $q = 1, \ldots, Q$
   2.3 for each $z_{n+1,q}(Y_j)$, $q \in \{1, \ldots, Q\}$,
      2.3.1 compute the kriging approximation $\tilde{f}_{n+1,j,q}$ on $\mathbb{X}_m$ from $\mathcal{I}_n \cup (Y_j, f(Y_j) = z_{n+1,q}(Y_j))$, using the weights $\lambda_i(Y_k; \{\underline{X}_n, Y_j\})$, $i = 1, \ldots, (n+1)$, $k = 1, \ldots, m$, obtained at Step 2.1.
      2.3.2 for each $k \in \{1, \ldots, m\}$, compute $v_{n+1}(Y_k; Y_j, z_{n+1,q}(Y_j))$, using u, $\tilde{f}_{n+1,j,q}$ obtained in 2.3.1, and $\sigma^2(Y_k; \{\underline{X}_n, Y_j\})$ obtained in 2.1
   2.4 compute $J_n(Y_j) = \sum_{k=1}^{m} \sum_{q=1}^{Q} w'_q \, v_{n+1}(Y_k; Y_j, z_{n+1,q}(Y_j))$.
3. find $j^\star = \operatorname{argmin}_j J_n(Y_j)$ and set $X_{n+1} = Y_{j^\star}$

Table 3 Sequential estimation of a probability of failure

1. Construct an initial design of size $n_0 < N$ and evaluate f at the points of the initial design.
2. Choose a Gaussian process ξ (in practice, this amounts to choosing a parametric form for the mean of ξ and a parametric covariance function k_θ)
3. Generate a Monte Carlo sample $\mathbb{X}_m = \{Y_1, \ldots, Y_m\}$ of size m from $\mathsf{P}_{\mathbb{X}}$
4. While the evaluation budget N is not exhausted,
   4.1 optional step: estimate the parameters of the covariance function (case of a plug-in approach);
   4.2 select a new evaluation point, using past evaluation results, the prior ξ and $\mathbb{X}_m$;
   4.3 perform the new evaluation.
5. Estimate the probability of failure obtained from the N evaluations of f (for instance, by using $\mathsf{E}_N(\alpha_m) = \frac{1}{m} \sum_j p_N(Y_j)$).

4 Other strategies proposed in the literature

4.1 Estimation of a probability of failure and closely related objectives

Given a real function f defined over $\mathbb{X} \subseteq \mathbb{R}^d$, and a threshold $u \in \mathbb{R}$, consider the following possible goals:

1. estimate a region $\Gamma \subset \mathbb{X}$ of the form $\Gamma = \{x \in \mathbb{X} \mid f(x) > u\}$;
2. estimate the level set $\partial\Gamma = \{x \in \mathbb{X} \mid f(x) = u\}$;
3. estimate f precisely in a neighborhood of $\partial\Gamma$;

4. estimate the probability of failure $\alpha = \mathsf{P}_{\mathbb{X}}(\Gamma)$ for a given probability measure $\mathsf{P}_{\mathbb{X}}$.

These different goals are, in fact, closely related: indeed, they all require, more or less explicitly, to select sampling points in order to get a fine knowledge of the function f in a neighborhood of the level set $\partial\Gamma$ (the location of which is unknown before the first evaluation). Any strategy proposed for one of the first three objectives is therefore expected to perform reasonably well on the fourth one, which is the topic of this paper.

Several strategies recently introduced in the literature are presented in Sections 4.2 and 4.3, and will be compared numerically to the SUR strategy in Section 5. Each of these strategies has been initially proposed by its authors to address one or several of the above objectives, but they will only be discussed in this paper from the point of view of their performance on the fourth one. Of course, a comparison focused on any other objective would probably be based on different performance metrics, and thus could yield a different performance ranking of the strategies.

4.2 The targeted IMSE criterion

The *targeted IMSE* proposed in Picheny et al. (2010) is a modification of the IMSE (Integrated Mean Square Error) sampling criterion (Sacks et al., 1989). While the IMSE sampling criterion computes the average of the kriging variance (over a compact domain $\mathbb{X}$) in order to achieve a space-filling design, the targeted IMSE computes a weighted average of the kriging variance for a better exploration of the regions near the frontier of the domain of failure, as in Oakley (2004). The idea is to put a large weight in regions where the kriging prediction is close to the threshold u, and a small one otherwise. Given $\mathcal{I}_n$, the targeted IMSE sampling criterion, hereafter abbreviated as tIMSE, can be written as

$$J_n^{\mathrm{tIMSE}}(x) = \mathsf{E}_n\left(\int_{\mathbb{X}} \left(\xi - \widehat{\xi}_{n+1}\right)^2 W_n \,\mathrm{d}\mathsf{P}_{\mathbb{X}} \;\middle|\; X_{n+1} = x\right) \tag{27}$$

$$= \int_{\mathbb{X}} \sigma^2\left(y; X_1, \ldots, X_n, x\right) W_n(y)\, \mathsf{P}_{\mathbb{X}}(\mathrm{d}y), \tag{28}$$

where W_n is a weight function based on $\mathcal{I}_n$. The weight function suggested by Picheny et al. (2010) is

$$W_n(x) = \frac{1}{s_n(x)\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{\widehat{\xi}_n(x) - u}{s_n(x)}\right)^2\right), \tag{29}$$

where $s_n^2(x) = \sigma_\varepsilon^2 + \sigma_n^2(x)$. Note that $W_n(x)$ is large when $\widehat{\xi}_n(x) \approx u$ and $\sigma_n^2(x) \approx 0$, i.e., when the function is known to be close to u.

The tIMSE criterion operates a trade-off between global uncertainty reduction (high kriging variance σ_n^2) and exploration of target regions (high weight function W_n). The weight function depends on a parameter $\sigma_\varepsilon > 0$, which allows to tune the width of the "window of interest" around the threshold. For large values of σ_ε, J^{tIMSE} behaves approximately like the IMSE sampling criterion. The choice of an appropriate value for σ_ε, when the goal is to estimate a probability of failure, will be discussed on the basis of numerical experiments in Section 5.3.

The tIMSE strategy requires a computation of the expectation with respect to $\xi(x)$ in (27), which can be done analytically, yielding (28). The computation of the integral with respect to $\mathsf{P}_{\mathbb{X}}$ on $\mathbb{X}$ can be carried out with a Monte Carlo approach, as explained in Section 3.3. Finally, the optimization of the criterion is replaced by a discrete search in our implementation.

4.3 Criteria based on the marginal distributions

Other sampling criteria proposed by Ranjan et al. (2008), Bichon et al. (2008) and Echard et al. (2010a,b) are briefly reviewed in this section[7]. A common feature of these three criteria is that, unlike the SUR and tIMSE criteria discussed so far, they only depend on the *marginal posterior distribution* at the considered candidate point $x \in \mathbb{X}$, which is a Gaussian $\mathcal{N}\big(\widehat{\xi}_n(x), \sigma_n^2(x)\big)$ distribution. As a consequence, they are of course much cheaper to compute than integral criteria like SUR and tIMSE.

A natural idea, in order to sequentially improve the estimation of the probability of failure, is to visit the point $x \in \mathbb{X}$ where the event $\{\xi(x) \geq u\}$ is the most uncertain. This idea, which has been explored by Echard, Gayton, and Lemaire (2010a,b), corresponds formally to the sampling criterion

$$J_n^{\mathrm{EGL}}(x) \;=\; \tau_n(x) \;=\; 1 - \Phi\left(\frac{\big|u - \widehat{\xi}_n(x)\big|}{\sigma_n(x)}\right). \tag{30}$$

As in the case of the tIMSE criterion and also, less explicitly, in SUR criteria, a trade-off is realized between global uncertainty reduction (choosing points with a high $\sigma_n^2(x)$) and exploration of the neighborhood of the estimated contour (where $\big|u - \widehat{\xi}_n(x)\big|$ is small).

The same leading principle motivates the criteria proposed by Ranjan et al. (2008) and Bichon et al. (2008), which can be seen as special cases of the following sampling criterion:

$$J_n^{\mathrm{RB}}(x) \;:=\; \mathsf{E}_n\left(\max\Big(0, \epsilon(x)^\delta - |u - \xi(x)|^\delta\Big)\right), \tag{31}$$

where $\epsilon(x) = \kappa\,\sigma_n(x)$, $\kappa, \delta > 0$. The following proposition provides some insights into this sampling criterion:

Proposition 4 *Define $G_{\kappa,\delta} :]0,1[\to \mathbb{R}_+$ by*

$$G_{\kappa,\delta}(p) \;:=\; \mathsf{E}\left(\max\Big(0, \kappa^\delta - \big|\Phi^{-1}(p) + U\big|\Big)\right),$$

where U is a Gaussian $\mathcal{N}(0,1)$ random variable. Let φ and Φ denote respectively the probability density function and the cumulative distribution function of U.

a) $G_{\kappa,\delta}(p) = G_{\kappa,\delta}(1-p)$ for all $p \in]0,1[$.

b) $G_{\kappa,\delta}$ is strictly increasing on $]0,1/2]$ and vanishes at 0. Therefore, $G_{\kappa,\delta}$ is also strictly decreasing on $[1/2,1[$, vanishes at 1, and has a unique maximum at $p = 1/2$.

c) Criterion (31) can be rewritten as

$$J_n^{RB}(x) = \sigma_n(x)^\delta\, G_{\kappa,\delta}\big(p_n(x)\big). \tag{32}$$

d) $G_{\kappa,1}$ has the following closed-form expression:

$$\begin{aligned} G_{\kappa,1}(p) \;=\; & \kappa\left(\Phi(t^+) - \Phi(t^-)\right) \\ & - t\left(2\Phi(t) - \Phi(t^+) - \Phi(t^-)\right) \\ & - \left(2\varphi(t) - \varphi(t^+) - \varphi(t^-)\right), \end{aligned} \tag{33}$$

where $t = \Phi^{-1}(1-p)$, $t^+ = t + \kappa$ and $t^- = t - \kappa$.

[7] Note that the paper of Ranjan et al. (2008) is the only one in this category that does not address the problem of estimating a probability of failure (i.e., Objective 4 of Section 4.1).

e) $G_{\kappa,2}$ *has the following closed-form expression:*

$$\begin{aligned} G_{\kappa,2}(p) \;=\; & \left(\kappa^2 - 1 - t^2\right)\left(\Phi(t^+) - \Phi(t^-)\right) \\ & - 2t\left(\varphi(t^+) - \varphi(t^-)\right) \\ & + t^+\varphi(t^+) - t^-\varphi(t^-), \end{aligned} \tag{34}$$

with the same notations.

It follows from a) and c) that $J_n^{\mathrm{RB}}(x)$ can also be seen as a function of the kriging variance $\sigma_n^2(x)$ and the probability of misclassification $\tau_n(x) = \min\left(p_n(x), 1 - p_n(x)\right)$. Note that, in the computation of $G_{\kappa,\delta}\big(p_n(x)\big)$, the quantity denoted by t in (33) and (34) is equal to $\big(u - \widehat{\xi}_n(x)\big)/\sigma_n(x)$, i.e., equal to the normalized distance between the predicted value and the threshold.

Bichon et al.'s *expected feasibility* function corresponds to (32) with $\delta = 1$, and can be computed efficiently using (33). Similarly, Ranjan et al.'s *expected improvement*[8] function corresponds to (32) with $\delta = 2$, and can be computed efficiently using (34). The proof of Proposition 4 is provided in Appendix B.

Remark 3 In the case $\delta = 1$, our result coincides with the expression given by Bichon et al. (2008, Eq. (17)). In the case $\delta = 2$, we have found and corrected a mistake in the computations of Ranjan et al. (2008, Eq. (8) and Appendix B).

5 Numerical experiments

5.1 A one-dimensional illustration of a SUR strategy

The objective of this section is to show the progress of a SUR strategy in a simple one-dimensional case. We wish to estimate $\alpha = \mathsf{P}_{\mathbb{X}}\{f > 1\}$, where $f : \mathbb{X} = \mathbb{R} \to \mathbb{R}$ is such that $\forall x \in \mathbb{R}$,

$$f(x) = (0.4x - 0.3)^2 + \exp\left(-11.534\,|x|^{1.95}\right) + \exp(-5(x - 0.8)^2)\,,$$

and where $\mathbb{X}$ is endowed with the probability distribution $\mathsf{P}_{\mathbb{X}} = \mathcal{N}(0, \sigma_{\mathbb{X}}^2)$, $\sigma_{\mathbb{X}} = 0.4$, as depicted in Figure 2. We know in advance that $\alpha \approx 0.2$. Thus, a Monte Carlo sample of size $m = 1500$ will give a good estimate of α.

In this illustration, ξ is a Gaussian process with constant but unknown mean and a Matérn covariance function, whose parameters are kept *fixed*, for the sake of simplicity. Figure 2 shows an initial design of four points and the sampling criterion $J_{1,n=4}^{\mathrm{SUR}}$. Notice that the sampling criterion is only computed at the points of the Monte Carlo sample. Figures 3 and 4 show the progress of the SUR strategy after a few iterations. Observe that the unknown function f is sampled so that the probability of excursion p_n almost equals zero or one in the region where the density of $\mathsf{P}_{\mathbb{X}}$ is high.

[8] Despite its name and some similarity between the formulas, this criterion should not be confused with the well-known EI criterion in the field of optimization (Mockus et al., 1978; Jones et al., 1998).

5.2 An example in structural reliability

In this section, we evaluate all criteria discussed in Section 3 and Section 4 through a classical benchmark example in structural reliability (see, e.g., Borri and Speranzini, 1997; Waarts, 2000; Schueremans, 2001; Deheeger, 2008). Echard et al. (2010a,b) used this benchmark to make a comparison among several methods proposed in Schueremans and Gemert (2005), some of which are based on the construction of a response surface. The objective of the benchmark is to estimate the probability of failure of a so-called *four-branch series system*. A failure happens when the system is working under the threshold $u = 0$. The performance function f for this system is defined as

$$f : (x_1, x_2) \in \mathbb{R}^2 \mapsto f(x_1, x_2) = \min \begin{Bmatrix} 3 + 0.1(x_1 - x_2)^2 - (x_1 + x_2)/\sqrt{2}; \\ 3 + 0.1(x_1 - x_2)^2 + (x_1 + x_2)/\sqrt{2}; \\ (x_1 - x_2) + 6/\sqrt{2}; \\ (x_2 - x_1) + 6/\sqrt{2} \end{Bmatrix} .$$

The uncertain input factors are supposed to be independent and have standard normal distribution. Figure 5 shows the performance function, the failure domain and the input distribution. Observe that f has a first-derivative discontinuity along four straight lines originating from the point $(0, 0)$.

For each sequential method, we will follow the procedure described in Table 3. We generate an initial design of $n_0 = 10$ points (five times the dimension of the factor space) using a maximin LHS (Latin Hypercube Sampling)[9] on $[-6; 6] \times [-6; 6]$. We choose a Monte Carlo sample of size $m = 30000$. Since the true probability of failure is approximately $\alpha = 0.4\%$ in this example, the coefficient of variation for α_m is $1/\sqrt{m\,\alpha} \approx 9\%$. The same initial design and Monte Carlo sample are used for all methods.

A Gaussian process with constant unknown mean and a Matérn covariance function is used as our prior information about f. The parameters of the Matérn covariance functions are estimated on the initial design by REML (see, e.g. Stein, 1999). In this experiment, we follow the common practice of re-estimating the parameters of the covariance function during the sequential strategy, but only once every ten iterations to save some computation time.

The probability of failure is estimated by (13). To evaluate the rate of convergence, we compute the number n_γ of iterations that must be performed using a given strategy to observe a stabilization of the relative error of estimation within an interval of length 2γ:

$$n_\gamma = \min \left\{ n \geq 0; \forall k \geq n, \frac{|\widehat{\alpha}_{n_0+k} - \alpha_m|}{\alpha_m} < \gamma \right\} .$$

All the available sequential strategies are run 100 times, with different initial designs and Monte Carlo samples. The results for $\gamma = 0.10$, $\gamma = 0.03$ and $\gamma = 0.01$ are summarized in Table 4. We shall consider that $n_{0.1}$ provides a measure of the performance of the strategy in the "initial phase", where a rough estimate of α is to be found, whereas $n_{0.03}$ and $n_{0.01}$ measure the performance in the "refinement phase".

The four variants of the SUR strategy (see Table 1) have been run with $Q = 12$ and either $m_0 = 10$ or $m_0 = 500$. The performance are similar for all four variants and for both values of m_0. It appears, however, that the criterions J_1^{SUR} and $J_2^{\mathrm{SUR}}2$ (i.e., the criterions given directly by Proposition 3) are slightly better than J_3^{SUR} and J_4^{SUR}; this will be confirmed by the simulations of Section 5.3. It also

[9] More precisely, we use Matlab's `lhsdesign()` function to select the best design according to the maximin criterion among 10^4 randomly generated LHS designs.

seems that the SUR algorithm is slightly slower to obtain a rough estimate of the probability of failure when m_0 is very small, but performs very well in the refinement phase. (Note that $m_0 = 10$ is a drastic pruning for a sample of size $m = 30000$.)

The tIMSE strategy has been run for three different values of its tuning parameter σ_ε^2, using the pruning scheme with $m_0 = 500$. The best performance is obtained for $\sigma_\varepsilon^2 \approx 0$, and is almost as good as the performance of SUR stragies with the same value of $m0$ (a small loss of performance, of about one evaluation on average, can be noticed in the refinement phase). Note that the required accuracy was not reached after 200 iterations in 17% of the runs for $\sigma_\varepsilon^2 = 1$. In fact, the tIMSE strategy tends to behave like a space-filling strategy in this case. Figure 6 shows the points that have been evaluated in three cases: the evaluations are less concentrated on the boundary between the safe and the failure region when $\sigma_\varepsilon^2 = 1$.

Finally, the results obtained for J^{RB} and J^{EGL} indicate that the corresponding strategies are clearly less efficient in the "initial phase" than strategies based on J_1^{SUR} or J_2^{SUR}. For $\gamma = 0.1$, the average loss with respect to J_1^{SUR} is between approximately 0.9 evaluations for the best case (criterion J^{RB} with $\delta = 2$, $\kappa = 2$) and 3.9 evaluations for the worst case. For $\gamma = 0.03$, the loss is between 1.4 evaluations (also for (criterion J^{RB} with $\delta = 2$, $\kappa = 2$) and 3.5 evaluations. This loss of efficiency can also be observed very clearly on the 90^{th} percentile in the inital phase. Criterion J^{RB} seems to perform best with $\delta = 2$ and $\kappa = 2$ in this experiment, but this will not be confirmed by the simulations of Section 5.3. Tuning the parameters of this criterion for the estimation of a probability of failure does not seem to be an easy task.

Table 4 Comparison of the convergence to α_m in the benchmark example Section 5.2 for different sampling strategies. The first number (bold text) is the average value of n_γ over 100 runs. The numbers between brackets indicate the 10^{th} and 90^{th} percentile.

criterion	parameters	$\gamma = 0.10$	$\gamma = 0.03$	$\gamma = 0.01$
J_1^{SUR}	$m_0 = 500$	**16.1** [10–22]	**25.7** [17–35]	**36.0** [26–48]
	$m_0 = 10$	**19.4** [11–28]	**28.1** [19–38]	**35.4** [26–44]
J_2^{SUR}	$m_0 = 500$	**16.4** [10–24]	**25.7** [19–33]	**35.5** [25–45]
	$m_0 = 10$	**20.0** [11–30]	**28.3** [20–39]	**35.3** [26–44]
J_3^{SUR}	$m_0 = 500$	**18.2** [10–27]	**26.9** [18–37]	**35.9** [27–46]
	$m_0 = 10$	**20.1** [11–30]	**28.0** [20–36]	**35.2** [25–44]
J_4^{SUR}	$m_0 = 500$	**17.2** [10–28]	**26.5** [20–36]	**35.2** [25–45]
	$m_0 = 10$	**21.4** [13–30]	**28.9** [20–38]	**35.5** [27–44]
J^{tIMSE}	$\sigma_\varepsilon^2 = 10^{-6}$	**16.6** [10–23]	**26.5** [19–36]	**37.3** [28–49]
	$\sigma_\varepsilon^2 = 0.1$	**15.9** [10–22]	**29.1** [19–43]	**50.5** [30–79]
	$\sigma_\varepsilon^2 = 1$	**21.7** [11–31]	**52.4** [31–85]	**79.5** [42–133](*)
J^{EGL}	–	**21.0** [11–31]	**29.2** [21–39]	**36.4** [28–44]
J^{RB}	$\delta = 1$, $\kappa = 0.5$	**18.7** [10–27]	**27.5** [20–35]	**36.6** [27–44]
	$\delta = 1$, $\kappa = 2.0$	**18.9** [11–28]	**28.3** [21–35]	**37.7** [30–45]
	$\delta = 2$, $\kappa = 0.5$	**17.6** [10–24]	**27.6** [20–34]	**37.1** [29–45]
	$\delta = 2$, $\kappa = 2.0$	**17.0** [10–21]	**27.1** [20–34]	**36.8** [29–44]

(*) The required accuracy was not reached after 200 iterations in 17% of the runs

Table 5 Size of the initial design and covariance parameters for the experiments of Section 5.3. The parametrization of the Matérn covariance function used here is defined in Appendix A.

d	n_0	σ^2	ν	ρ
1	3	1.0	2.0	0.100
2	10	1.0	2.0	0.252
3	15	1.0	2.0	0.363

5.3 Average performance on sample paths of a Gaussian process

This section provides a comparison of all the criteria introduced or recalled in this paper, on the basis of their average performance on the sample paths of a zero-mean Gaussian process defined on $\mathbb{X} = [0,1]^d$, for $d \in \{1,2,3\}$. In all experiments, the same covariance function is used for the generation of the sample paths and for the computation of the sampling criteria. We have considered isotropic Matérn covariance functions, whose parameters are given in Table 5. An initial maximin LHS design of size n_0 (also given in the table) is used: note that the value of n reported on the x-axis of Figures 7–11 is the total number of evaluations, including the initial design.

The d input variables are assumed to be independent and uniformly distributed on $[0,1]$, i.e., $\mathsf{P}_{\mathbb{X}}$ is the uniform distribution on $\mathbb{X}$. An m-sample $Y_1, \ldots, Y_m$ from $\mathsf{P}_{\mathbb{X}}$ is drawn one and for all, and used both for the approximation of integrals (in SUR and tIMSE criteria) and for the discrete search of the next sampling point (for all criteria). We take $m = 500$ and use the same MC sample for all criteria in a given dimension d.

We adopt the meta-estimation framework as described in Section 3.3; in other words, our goal is to estimate the MC estimator α_m. We choose to adjust the threshold u in order to have $\alpha_m = 0.02$ for all sample paths (note that, as a consequence, there are exactly $m\alpha_m = 10$ points in the failure region) and we measure the performance of a strategy after n evaluations by its relative mean-square error (MSE) expressed in decibels (dB):

$$\mathrm{rMSE} \;:=\; 10\,\log_{10}\left(\frac{1}{L}\sum_{l=1}^{L}\frac{\left(\widehat{\alpha}_{m,n}^{(l)}-\alpha_m\right)^2}{\alpha_m^2}\right),$$

where $\widehat{\alpha}_{m,n}^{(l)} = \frac{1}{m}\sum_{j=1}^{m} p_n^{(l)}(Y_j)$ is the posterior mean of the MC estimator α_m after n evaluations on the l^{th} simulated sample path ($L = 4000$).

We use a sequential maximin strategy as a reference in all of our experiments. This simple space-filling strategy is defined by $X_{n+1} = \mathrm{argmax}_j \, \min_{1\le i\le n} \left|Y_j - X_i\right|$, where the argmax runs over all indices j such that $Y_j \notin \{X_1, \ldots, X_n\}$. Note that this strategy does not depend on the choice of a Gaussian process model.

Our first experiment (Figure 7) provides a comparison of the four SUR strategies proposed in Section 3.2. It appears that all of them perform roughly the same when compared to the reference strategy. A closer look, however, reveals that the strategies J_1^{SUR} and J_2^{SUR} provided by Proposition 3 perform slightly better than the other two (noticeably so in the case $d = 3$).

The performance of the tIMSE strategy is shown on Figure 8 for several value of its tuning parameter σ_ε^2 (other values, not shown here, have been tried as well). It is clear that the performance of this strategy improves when σ_ε^2 goes to zero, whatever the dimension.

The performance of the strategy based on $J^{\mathrm{RB}}_{\kappa,\delta}$ is shown on Figure 9 for several values of its parameters. It appears that the criterion proposed by Bichon et al. (2008), which corresponds to $\delta = 1$, performs better than the one proposed by Ranjan et al. (2008), which corresponds to $\delta = 2$, for the same value of κ. Moreover, the value $\kappa = 0.5$ has been found in our experiments to produce the best results.

Figure 10 illustrates that the loss of performance associated to the "pruning trick" introduced in Section 3.4 can be negligible if the size m_0 of the pruned MC sample is large enough (here, m_0 has been taken equal to 50). In practice, the value of m_0 should be chosen small enough to keep the overhead of the sequential strategy reasonable—in other words, large values of m_0 should only be used for very complex computer codes.

Finally, a comparison involving the best strategy obtained in each category is presented on Figure 11. The best result is consistently obtained with the SUR strategy based on $J^{\mathrm{SUR}}_{1,n}$. The tIMSE strategy with $\sigma^2_\varepsilon \approx 0$ provides results which are almost as good. Note that both strategies are one-step lookahead strategies based on the approximation of the risk by an integral criterion, which makes them rather expensive to compute. Simpler strategies based on the marginal distribution (criteria J^{RB}_n and J^{EGL}_n) provide interesting alternatives for moderately expensive computer codes: their performances, although not as good as those of one-step lookahead criterions, are still much better than that of the reference space-filling strategy.

6 Concluding remarks

One of the main objectives of this paper was to present a synthetic viewpoint on sequential strategies based on a Gaussian process model and kriging for the estimation of a probability of failure. The starting point of this presentation is a Bayesian decision-theoretic framework from which the theoretical form of an optimal strategy for the estimation of a probability of failure can be derived. Unfortunately, the dynamic programming problem corresponding to this strategy is not numerically tractable. It is nonetheless possible to derive from there the ingredients of a sub-optimal strategy: the idea is to focus on one-step lookahead suboptimal strategies, where the exact risk is replaced by a substitute risk that accounts for the information gain about α expected from a new evaluation. We call such a strategy a *stepwise uncertainty reduction* (SUR) strategy. Our numerical experiments show that SUR strategies perform better, on average, than the other strategies proposed in the literature. However, this comes at a higher computational cost than strategies based only on marginal distributions. The tIMSE sampling criterion, which seems to have a convergence rate comparable to that of the SUR criterions when $\sigma^2_\varepsilon \approx 0$, also has a high computational complexity.

In which situations can we say that the sequential strategies presented in this paper are interesting alternatives to classical importance sampling methods for estimating a probability of failure, for instance the subset sampling method of Au and Beck (2001)? In our opinion, beyond the obvious role of the simulation budget N, the answer to this question depends on our capacity to elicit an appropriate prior. In the example of Section 5.2, as well as in many other examples of the literature using Gaussian processes in the domain of computer experiments, the prior is easy to choose because $\mathbb{X}$ is a low-dimensional space and f tends to be smooth. Then, the plug-in approach which consists in using ML or REML to estimate the parameters of the covariance function of the Gaussian process after each new evaluation is likely to succeed. If $\mathbb{X}$ is high-dimensional and f is expensive to evaluate, difficulties arise. In particular, our

sampling strategies do not take into account our uncertain knowledge of the covariance parameters, and there is no guarantee that ML estimation will do well when the points are chosen by a sampling strategy that favors some localized target region (the neighboorhood the frontier of the domain of failure in this paper, but the question is equally relevant in the field optimization, for instance). The difficult problem of deciding the size n_0 of the initial design is crucial in this connection. Fully Bayes procedures constitute a possible direction for future research, as long as they don't introduce an unacceptable computational overhead. Whatever the route, we feel that the robustness of Gaussian process-based sampling strategies with respect to the procedure of estimation of the covariance parameters should be addressed carefully in order to make these methods usable in the industrial world.

Software. We would like to draw the reader's attention on the recently published package KrigInv (Picheny and Ginsbourger, 2011) for the statistical computing environment R (see Hornik, 2010). This package provides an open source (GPLv3) implementation of all the strategies proposed in this paper. Please note that the simulation results presented in this paper were not obtained using this package, that was not available at the time of its writing.

Acknowledgements The research of Julien Bect, Ling Li and Emmanuel Vazquez was partially funded by the French *Agence Nationale de la Recherche* (ANR) in the context of the project OPUS (ref. ANR-07-CIS7-010) and by the French *pôle de compétitivité* SYSTEMATIC in the context of the project CSDL. David Ginsbourger acknowledges support from the French *Institut de Radioprotection et de Sûreté Nucléaire* (IRSN) and warmly thanks Yann Richet.

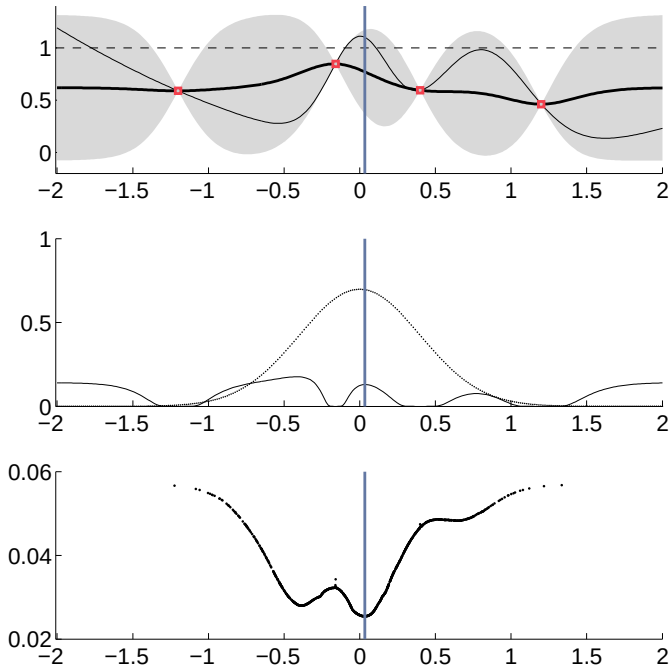

Fig. 2 Illustration of a SUR strategy. This figure shows the initial design. Top: threshold $u = 1$ (horizontal dashed line); function f (thin line); $n = 4$ initial evaluations (squares); kriging approximation f_n (thick line); 95% confidence intervals computed from the kriging variance (shaded area). Middle: probability of excursion (solid line); probability density of $\mathsf{P}_{\mathbb{X}}$ (dotted line). Bottom: graph of $J^{\mathrm{SUR}}_{1,n=4}(Y_i)$, $i = 1, \ldots, m = 1500$, the minimum of which indicates where the next evaluation of f should be done (i.e., near the origin).

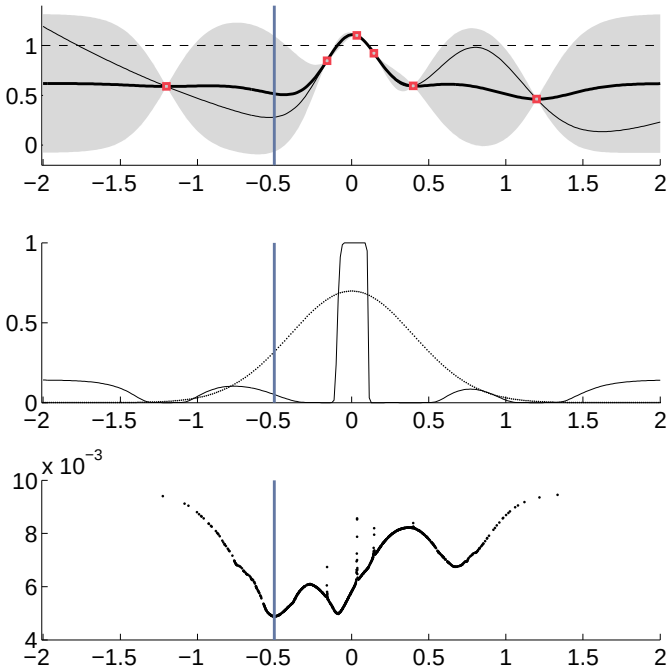

Fig. 3 Illustration of a SUR strategy (see also Figures 2 and 4). This figure shows the progress of the SUR strategy after two iterations—a total of $n = 6$ evaluations (squares) have been performed. The next evaluation point will be approximately at $x = -0.5$

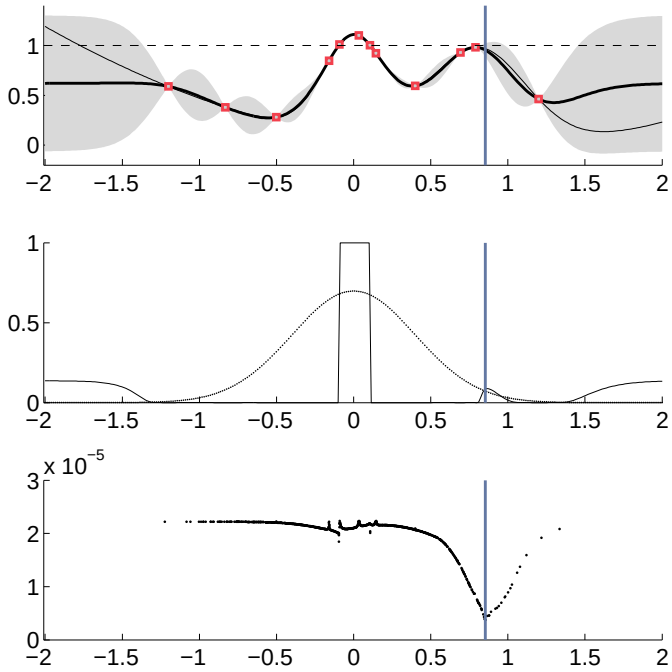

Fig. 4 Illustration of a SUR strategy (see also Figures 2 and 3). This figure shows the progress of the SUR strategy after eight iterations—a total of $n = 12$ evaluations (squares) have been performed. At this stage, the probability of excursion p_n almost equals 0 or 1 in the region where the density of $\mathsf{P}_{\mathbb{X}}$ is high.

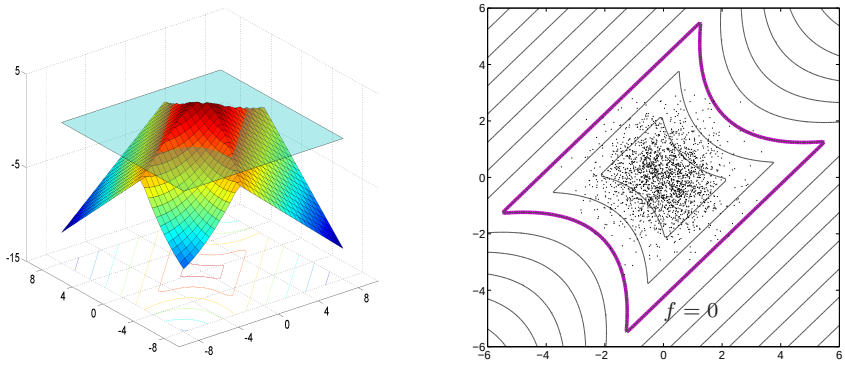

Fig. 5 Left: mesh plot of the performance function f corresponding to the four-branch series system; a failure happens when f is below the transparent plane; Right: contour plot of f; limit state $f = 0$ (thick line); sample of size $m = 3 \times 10^3$ from $\mathsf{P}_{\mathbb{X}}$ (dots).

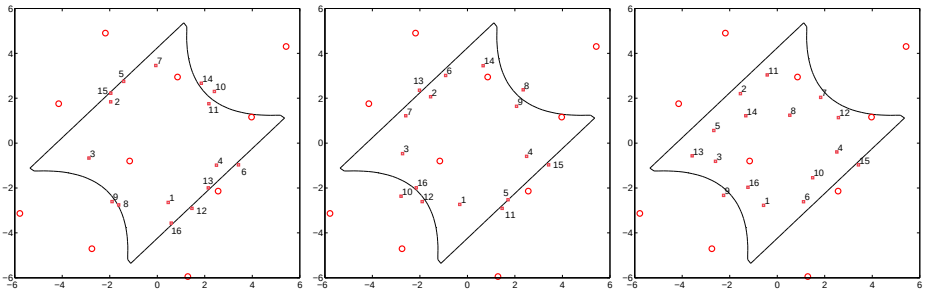

Fig. 6 The first 16 points (squares) evaluated using sampling criterion J_1^{SUR} (left), J^{tIMSE} with $\sigma_\varepsilon^2 = 0.1$ (middle), J^{tIMSE} with $\sigma_\varepsilon^2 = 1$ (right). Numbers near squares indicate the order of evaluation. The location of the $n_0 = 10$ points of the initial design are indicated by circles.

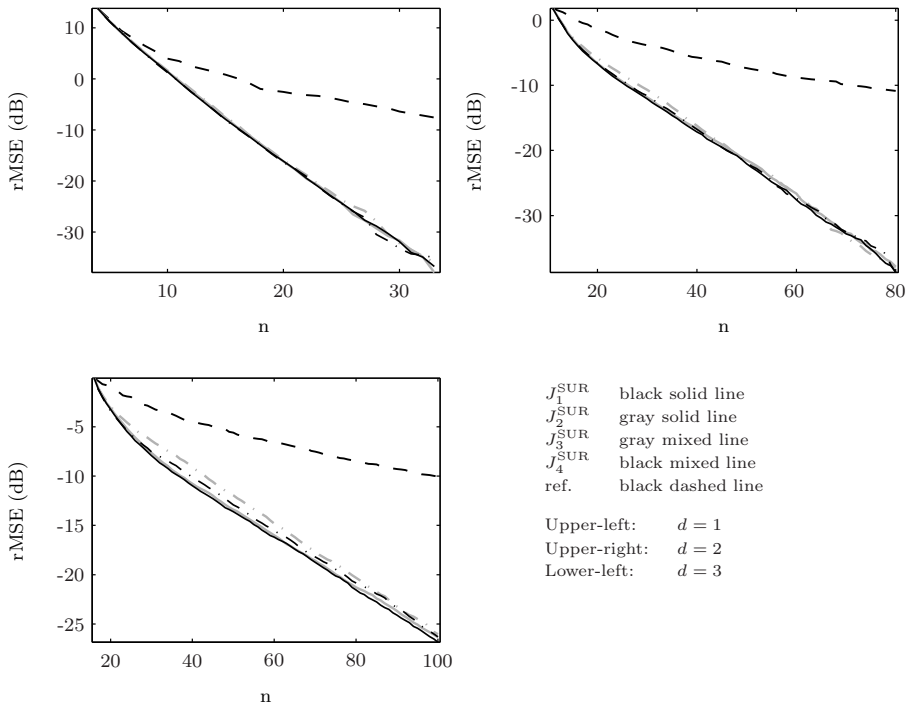

Fig. 7 Relative MSE performance of several SUR strategies.

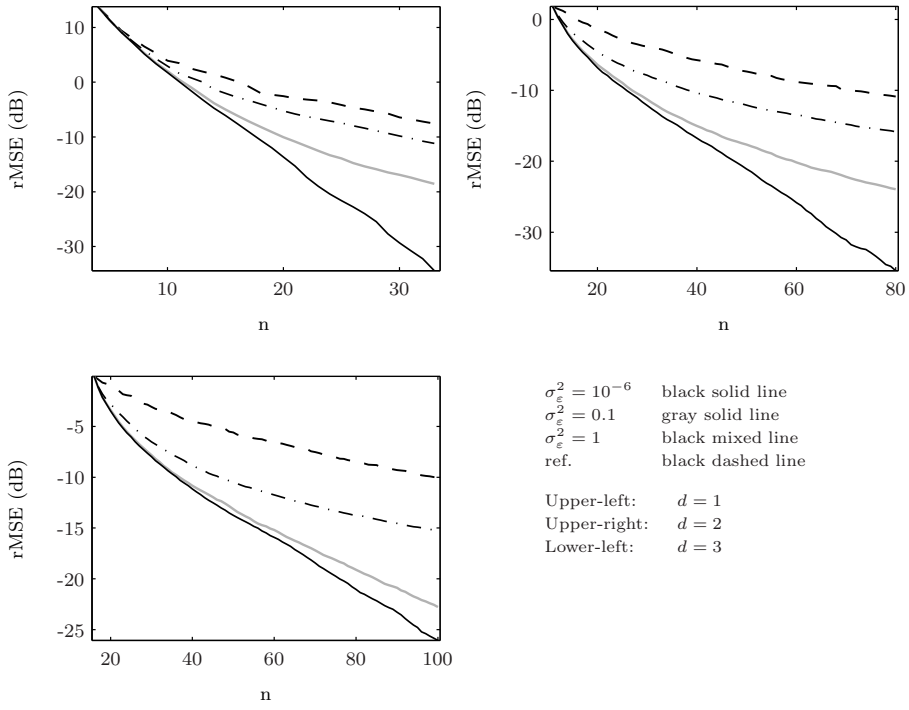

Fig. 8 Relative MSE performance of the tIMSE strategy for several values of its parameter.

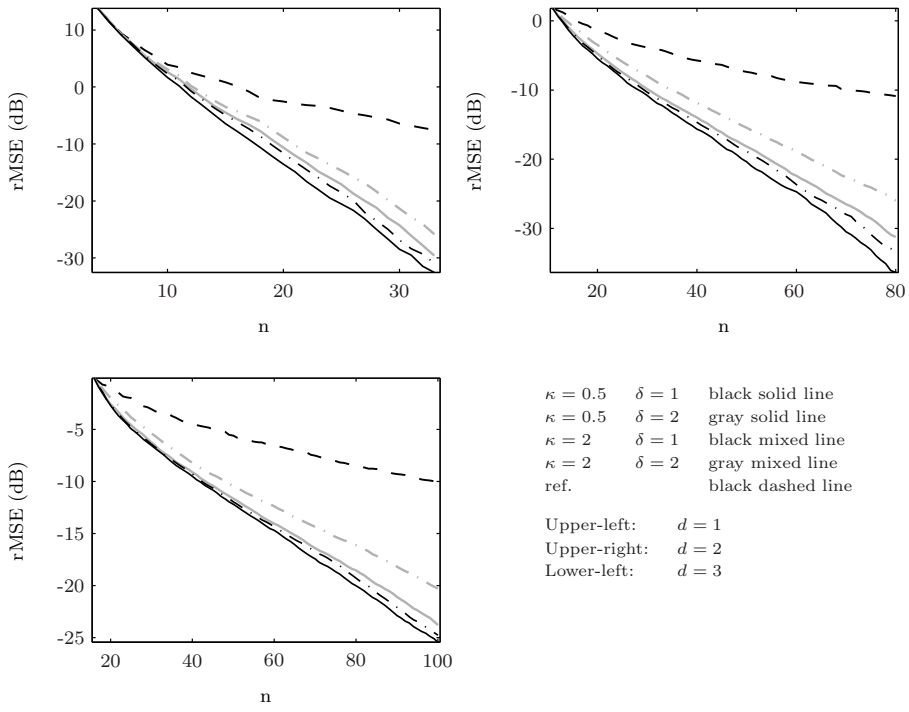

Fig. 9 Relative MSE performance of the J^{RB} criterion, for several values of its parameters.

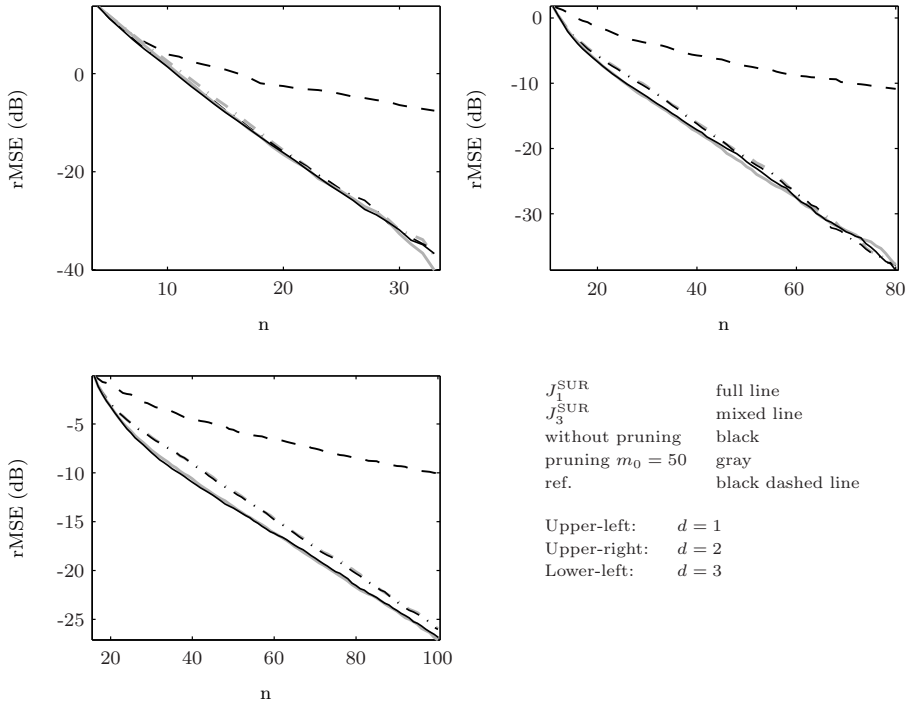

Fig. 10 Relative MSE performance of two SUR criteria, with and without the "pruning trick" described in Section 3.4. The black and gray lines are almost surimposed for each of the criterions J_1^{SUR} and J_3^{SUR}.

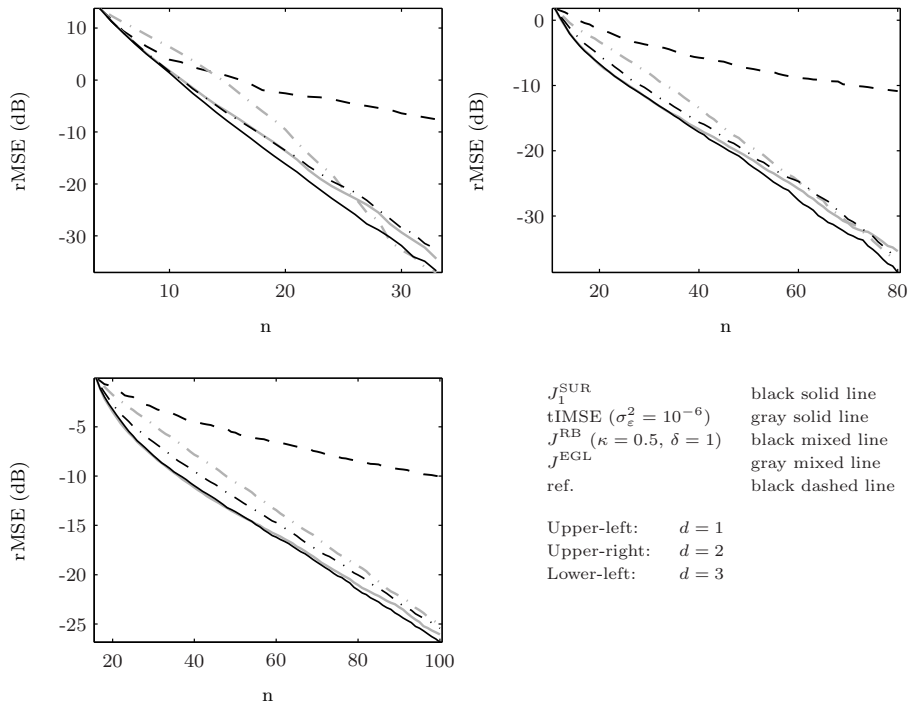

Fig. 11 Relative MSE performance the best strategy in each category.

Appendix

A The Matérn covariance

The exponential covariance and the Matérn covariance are among the most conventionally used stationary covariances in the literature of design and analysis of computer experiments. The Matérn covariance class (Yaglom, 1986) offers the possibility to adjust the regularity of ξ with a single parameter. Stein (1999) advocates the use of the following parametrization of the Matérn function:

$$\kappa_\nu(h) = \frac{1}{2^{\nu-1}\Gamma(\nu)} \left(2\nu^{1/2}h\right)^\nu \mathcal{K}_\nu\left(2\nu^{1/2}h\right), \quad h \in \mathbb{R} \tag{35}$$

where Γ is the Gamma function and $\mathcal{K}_\nu$ is the modified Bessel function of the second kind. The parameter $\nu > 0$ controls regularity at the origin of the function. To model a real-valued function f defined over $\mathbb{X} \subset \mathbb{R}^d$, with $d \geq 1$, we use the following anisotropic form of the Matérn covariance:

$$k_\theta(x,y) = \sigma^2 \kappa_\nu\left(\sqrt{\sum_{i=1}^d \frac{(x_{[i]} - y_{[i]})^2}{\rho_i^2}}\right), \quad x, y \in \mathbb{R}^d \tag{36}$$

where $x_{[i]}, y_{[i]}$ denote the i^{th} coordinate of x and y, the positive scalar σ^2 is a variance parameter (we have $k_\theta(x,x) = \sigma^2$), and the positive scalars ρ_i represent scale or *range* parameters of the covariance, *i.e.*, characteristic correlation lengths. Since $\sigma^2 > 0, \nu > 0, \rho_i > 0$, $i = 1, \ldots, d$, we can take the logarithm of these scalars, and consider the vector of parameters $\theta = \{\log\sigma^2, \log\nu, -\log\rho_1, \ldots, -\log\rho_d\} \in \mathbb{R}^{d+2}$, which is a practical parameterization when σ^2, ν, ρ_i, $i = 1, \ldots, d$, need to be estimated from data.

B Proof of Proposition 4

a) Using the identity $\Phi^{-1}(1-p) = -\Phi^{-1}(p)$, we get

$$\left|U + \Phi^{-1}(1-p)\right| \;=\; \left|U - \Phi^{-1}(p)\right| \;\stackrel{\mathrm{d}}{=}\; \left|U + \Phi^{-1}(p)\right|,$$

where $\stackrel{\mathrm{d}}{=}$ denotes an equality in distribution. Therefore $G_{\kappa,\delta}(1-p) = G_{\kappa,\delta}(p)$.

b) Let $S_p = \max\left(0, \kappa^\delta - \left|\Phi^{-1}(p) + U\right|\right)$. Straightforward computations show that $t \mapsto \mathsf{P}\left(|t+U| \leq v\right)$ is strictly decreasing to 0 on $[0, +\infty[$, for all $v > 0$. As a consequence, $p \mapsto \mathsf{P}\left(S_p < s\right)$ is strictly increasing to 1 on $[1/2, 1[$, for all $s \in \left]0, \kappa^\delta\right[$. Therefore, $G_{\kappa,\delta}$ is strictly decreasing on $[1/2, 1[$ and tends to zeros when $p \to 1$. The other assertions then follow from a).

c) Recall that $\xi(x) \sim \mathcal{N}\left(\widehat{\xi}_n(x), \sigma_n^2(x)\right)$ under P_n. Therefore $U := \left(\xi(x) - \widehat{\xi}_n(x)\right)/\sigma_n(x) \sim \mathcal{N}(0,1)$ under P_n, and the result follows by substitution in (31).

The closed-form expressions of Ranjan et al.'s and Bichon and al.'s criteria (assertions d) and e)) is established in the following sections.

B.1 A preliminary decomposition common to both criteria

Recall that $t = \Phi^{-1}(1-p)$, $t^+ = t + \kappa$ and $t^- = t - \kappa$. Then,

$$\begin{aligned} G_{\kappa,\delta}(p) &= G_{\kappa,\delta}(1-p) = \mathsf{E}\left(\max\left(0, \kappa^\delta - \left|t - U\right|^\delta\right)\right) \\ &= \int_{\kappa^\delta - |t-u|^\delta \geq 0} \left(\kappa^\delta - |t-u|^\delta\right) \varphi(u)\,\mathrm{d}u \\ &= \int_{t^-}^{t^+} \left(\kappa^\delta - |t-u|^\delta\right) \varphi(u)\,\mathrm{d}u \\ &= \kappa^\delta \left(\Phi(t^+) - \Phi(t^-)\right) - \underbrace{\int_{t^-}^{t^+} |t-u|^\delta \varphi(u)\,\mathrm{d}u}_{\text{Term } A}. \end{aligned} \tag{37}$$

The computation of the integral A will be carried separately in the next two sections for $\delta = 1$ and $\delta = 2$. For this purpose, we shall need the following elementary results:

$$\int_a^b u\varphi(u)\mathrm{d}u = \varphi(a) - \varphi(b)\,, \tag{38}$$

$$\int_a^b u^2\varphi(u)\mathrm{d}u = a\varphi(a) - b\varphi(b) + \Phi(b) - \Phi(a)\,. \tag{39}$$

B.2 Case $\delta = 1$

Let us compute the value A_1 of the integral A for $\delta = 1$:

$$\begin{aligned} A_1 &= \int_{t^-}^{t^+} |t-u|\,\varphi(u)\mathrm{d}u = \int_{t^-}^{t} (t-u)\varphi(u)\mathrm{d}u + \int_{t}^{t^+} (u-t)\varphi(u)\mathrm{d}u \\ &= t\left(\int_{t^-}^{t} \varphi(u)\,\mathrm{d}u - \int_{t}^{t^+} \varphi(u)\,\mathrm{d}u\right) - \int_{t^-}^{t} u\varphi(u)\,\mathrm{d}u + \int_{t}^{t^+} u\varphi(u)\,\mathrm{d}u \\ &= t\left(2\Phi(t) - \Phi(t^-) - \Phi(t^+)\right) + 2\varphi(t) - \varphi(t^-) - \varphi(t^+)\,, \end{aligned} \tag{40}$$

where (38) has been used to get the final result. Plugging (40) into (37) yields (33).

B.3 Case $\delta = 2$

Let us compute the value A_2 of the integral A for $\delta = 2$:

$$\begin{aligned} A_2 &= \int_{t^-}^{t^+} (t-u)^2\varphi(u)\,\mathrm{d}u \\ &= t^2\int_{t^-}^{t^+} \varphi(u)\,\mathrm{d}u - 2t\int_{t^-}^{t^+} u\varphi(u)\,\mathrm{d}u + \int_{t^-}^{t^+} u^2\varphi(u)\,\mathrm{d}u \\ &= t^2\left(\Phi(t^+) - \Phi(t^-)\right) - 2t\left(\varphi(t^-) - \varphi(t^+)\right) \\ &\quad + t^-\varphi(t^-) - t^+\varphi(t^+) + \Phi(t^+) - \Phi(t^-)\,, \end{aligned} \tag{41}$$

where (38) and (39) have been used to get the final result. Plugging (40) into (37) yields (34).

References

Arnaud, A., Bect, J., Couplet, M., Pasanisi, A., Vazquez, E.: Évaluation d'un risque d'inondation fluviale par planification séquentielle d'expériences. In: 42èmes Journées de Statistique (2010)

Au, S.K., Beck, J.: Estimation of small failure probabilities in high dimensions by subset simulation. Probab. Engrg. Mechan. **16**(4), 263–277 (2001)

Bayarri, M.J., Berger, J.O., Paulo, R., Sacks, J., Cafeo, J.A., Cavendish, J., Lin, C.H., Tu, J.: A framework for validation of computer models. Technometrics **49**(2), 138–154 (2007)

Berry, D.A., Fristedt, B.: Bandit problems: sequential allocation of experiments. Chapman & Hall (1985)

Bertsekas, D.P.: Dynamic programming and optimal control vol. 1. Athena Scientific (1995)

Bichon, B.J., Eldred, M.S., Swiler, L.P., Mahadevan, S., McFarland, J.M.: Efficient global reliability analysis for nonlinear implicit performance functions. AIAA Journal **46**(10), 2459–2468 (2008)

Bjerager, P.: On computational methods for structural reliability analysis. Structural Safety **9**, 76–96 (1990)

Borri, A., Speranzini, E.: Structural reliability analysis using a standard deterministic finite element code. Structural Safety **19**(4), 361–382 (1997)

Box, G.E.P., Draper, N.R.: Empirical Model-Building and Response Surfaces. Wiley (1987)

Bucher, C.G., Bourgund, U.: A fast and efficient response surface approach for structural reliability problems. Structural Safety **7**(1), 57–66 (1990)

Chilès, J.P., Delfiner, P.: Geostatistics: Modeling Spatial Uncertainty. Wiley, New York (1999)

Currin, C., Mitchell, T., Morris, M., Ylvisaker, D.: Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments. J. Amer. Statist. Assoc. **86**(416), 953–963 (1991)

Deheeger, F.: Couplage mécano-fiabiliste : ^{2}SMART – méthodologie d'apprentissage stochastique en fiabilité. Ph.D. thesis, Université Blaise Pascal – Clermont II (2008)

Deheeger, F., Lemaire, M.: Support vector machine for efficient subset simulations: ^{2}SMART method. In: Kanda, J., Takada, T., Furuta, H. (eds.) 10th International Conference on Application of Statistics and Probability in Civil Engineering, Proceedings and Monographs in Engineering, Water and Earth Sciences, pp. 259–260. Taylor & Francis (2007)

Echard, B., Gayton, N., Lemaire, M.: Kriging-based Monte Carlo simulation to compute the probability of failure efficiently: AK-MCS method. In: 6èmes Journées Nationales de Fiabilité, 24–26 mars, Toulouse, France (2010a)

Echard, B., Gayton, N., Lemaire, M.: Structural reliability assessment using kriging metamodel and active learning. In: IFIP WG 7.5 Working Conference on Reliability and Optimization of Structural Systems (2010b)

Fleuret, F., Geman, D.: Graded learning for object detection. In: Proceedings of the workshop on Statistical and Computational Theories of Vision of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR/SCTV) (1999)

Frazier, P.I., Powell, W.B., Dayanik, S.: A knowledge-gradient policy for sequential information collection. SIAM Journal on Control and Optimization **47**(5), 2410–2439 (2008)

Ginsbourger, D.: Métamodèles multiples pour l'approximation et l'optimisation de fonctions numériques multivariables. Ph.D. thesis, Ecole nationale supérieure des Mines de Saint-Etienne (2009)

Ginsbourger, D., Le Riche, R., L., C.: Kriging is well-suited to parallelize optimization. In: Hiot, L.M., Ong, Y.S., Tenne, Y., Goh, C.K. (eds.) Computational Intelligence in Expensive Optimization Problems, *Adaptation Learning and Optimization*, vol. 2, pp. 131–162. Springer (2010)

Handcock, M.S., Stein, M.L.: A bayesian analysis of kriging. Technometrics **35**(4), 403–410 (1993)

Hornik, K.: The R FAQ (2010). URL `http://CRAN.R-project.org/doc/FAQ/R-FAQ.html`. ISBN 3-900051-08-9

Hurtado, J.E.: An examination of methods for approximating implicit limit state functions from the viewpoint of statistical learning theory. Structural Safety **26**(3), 271–293 (2004)

Hurtado, J.E.: Filtered importance sampling with support vector margin: A powerful method for structural reliability analysis. Structural Safety **29**(1), 2–15 (2007)

Jones, D.R., Schonlau, M., William, J.: Efficient global optimization of expensive black-box functions. Journal of Global Optimization **13**(4), 455–492 (1998)

Kennedy, M., O'Hagan, A.: Bayesian calibration of computer models. Journal of the Royal Statistical Society. Series B (Statistical Methodology) **63**(3), 425–464 (2001)

Kimeldorf, G.S., Wahba, G.: A correspondence between Bayesian estimation on stochastic processes and smoothing by splines. Ann. Math. Statist. **41**(2), 495–502 (1970)

Kushner, H.J.: A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. J. Basic Engineering **86**, 97–106 (1964)

Loeppky, J.L., Sacks, J., Welch, W.J.: Choosing the sample size of a computer experiment: A practical guide. Technometrics **51**(4), 366–376 (2009)

Mockus, J.: Bayesian Approach to Global Optimization. Theory and Applications. Kluwer Academic Publisher, Dordrecht (1989)

Mockus, J., Tiesis, V., Zilinskas, A.: The application of Bayesian methods for seeking the extremum. In: Dixon, L., Szego, E.G. (eds.) Towards Global Optimization, vol. 2, pp. 117–129. Elsevier (1978)

Oakley, J.: Estimating percentiles of uncertain computer code outputs. J. Roy. Statist. Soc. Ser. C **53**(1), 83–93 (2004)

Oakley, J., O'Hagan, A.: Bayesian inference for the uncertainty distribution of computer model outputs. Biometrika **89**(4) (2002)

Oakley, J., O'Hagan, A.: Probabilistic sensitivity analysis of complex models: a Bayesian approach. Journal of the Royal Statistical Society: Series B (Statistical Methodology) **66**(3), 751–769 (2004)

O'Hagan, A.: Curve fitting and optimal design for prediction. Journal of the Royal Statistical Society. Series B (Methodological) **40**(1), 1–42 (1978)

Parzen, E.: An approach to time series analysis. Ann. Math. Stat. **32**, 951–989 (1962)

Paulo, R.: Default priors for gaussian processes. Annals of Statistics **33**(2), 556–582 (2005)

Picheny, V., Ginsbourger, D.: KrigInv : Kriging-based inversion for deterministic and noisy computer experiments, version 1.1 (2011). URL http://cran.r-project.org/web/packages/KrigInv

Picheny, V., Ginsbourger, D., Roustant, O., Haftka, R.T., Kim, N.H.: Adaptive designs of experiments for accurate approximation of target regions (2010)

Piera-Martinez, M.: Modélisation des comportements extrêmes en ingénierie. Ph.D. thesis, Université Paris Sud - Paris XI (2008)

Pradlwarter, H., Schuëller, G., Koutsourelakis, P., Charmpis, D.: Application of line sampling simulation method to reliability benchmark problems. Structural Safety **29**(3), 208 – 221 (2007). A Benchmark Study on Reliability in High Dimensions

Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical Recipes in C. The Art of Scientific Computing (Second Edition). Cambridge University Press (1992)

Rajashekhar, M.R., Ellingwood, B.R.: A new look at the response surface approach for reliability analysis. Structural Safety **12**(3), 205–220 (1993)

Ranjan, P., Bingham, D., Michailidis, G.: Sequential experiment design for contour estimation from complex computer codes. Technometrics **50**(4), 527–541 (2008)

Rubinstein, R., Kroese, D.: The Cross-Entropy Method. Springer (2004)

Sacks, J., Welch, W.J., Mitchell, T.J., Wynn, H.P.: Design and analysis of computer experiments. Statistical Science **4**(4), 409–435 (1989)

Santner, T.J., Williams, B.J., Notz, W.: The Design and Analysis of Computer Experiments. Springer Verlag (2003)

Schueremans, L.: Probabilistic evaluation of structural unreinforced masonry. Ph.D. thesis, Catholic University of Leuven (2001)

Schueremans, L., Gemert, D.V.: Benefit of splines and neural networks in simulation based structural reliability analysis. Structural safety **27**(3), 246–261 (2005)

Stein, M.L.: Interpolation of Spatial Data: Some Theory for Kriging. Springer, New York (1999)

Vazquez, E., Bect, J.: A sequential Bayesian algorithm to estimate a probability of failure. In: Proceedings of the 15th IFAC Symposium on System Identification, SYSID 2009 15th IFAC Symposium on System Identification, SYSID 2009. Saint-Malo France (2009)

Vazquez, E., Piera-Martinez, M.: Estimation du volume des ensembles d'excursion d'un processus gaussien par krigeage intrinsèque. In: 39ème Journées de Statistiques Conférence Journée de Statistiques. Angers France (2007)

Vestrup, E.M.: The Theory of Measures and Integration. Wiley (2003)

Villemonteix, J.: Optimisation de fonctions coûteuses. Ph.D. thesis, Université Paris-Sud XI, Faculté des Sciences d'Orsay (2008)

Villemonteix, J., Vazquez, E., Walter, E.: An informational approach to the global optimization of expensive-to-evaluate functions. Journal of Global Optimization **44**(4), 509–534 (2009)

Waarts, P.H.: Structural reliability using finite element methods: an appraisal of DARS. Ph.D. thesis, Delft University of Technology (2000)

Welch, W.J., Buck, R.J., Sacks, J., Wynn, H.P., Mitchell, T.J., Morris, M.D.: Screening, predicting and computer experiments. Technometrics **34**, 15–25 (1992)

Yaglom, A.M.: Correlation Theory of Stationary and Related Random Functions I: Basic Results. Springer Series in Statistics. Springer-Verlag, New york (1986)